

**VALIDACIÓN Y PUESTA EN MARCHA DEL ALGORITMO DE CONTROL DE
TRÁFICO MULTINODAL BASADO EN INTELIGENCIA COMPUTACIONAL Y SU
INTEGRACIÓN A LA CENTRAL DE TRÁFICO**

CARLOS EDUARDO TRIANA RUBIO

**UNIVERSIDAD AUTÓNOMA DE OCCIDENTE
FACULTAD DE INGENIERÍA
DEPARTAMENTO DE AUTOMÁTICA Y ELECTRÓNICA
PROGRAMA DE INGENIERÍA MECATRÓNICA
SANTIAGO DE CALI
2013**

**VALIDACIÓN Y PUESTA EN MARCHA DEL ALGORITMO DE CONTROL DE
TRÁFICO MULTINODAL BASADO EN INTELIGENCIA COMPUTACIONAL Y SU
INTEGRACIÓN A LA CENTRAL DE TRÁFICO**

CARLOS EDUARDO TRIANA RUBIO

**Pasantía Institucional para optar al título de
Ingeniero Mecatrónico**

**Director
JUAN CARLOS MENA MORENO
Ingeniero Electricista**

**UNIVERSIDAD AUTÓNOMA DE OCCIDENTE
FACULTAD DE INGENIERÍA
DEPARTAMENTO DE AUTOMÁTICA Y ELECTRÓNICA
PROGRAMA DE INGENIERÍA MECATRÓNICA
SANTIAGO DE CALI
2013**

Nota de Aceptación:

**Aprobado por el comité de Grado
en cumplimiento de los requisitos
exigidos por la Universidad
Autónoma de Occidente para optar
al título de Ingeniero Mecatrónico.**

JESÚS ALFONSO LÓPEZ SOTELO
Jurado

JIMMY TOMBÉ ANDRADE
Jurado

Marzo 21 de 2013

Dedico mi Trabajo de Grado a:

Dios, porque ha sido quien me ha guiado a lo largo de toda mi carrera y me ha dado las herramientas, capacidades, sabiduría e inteligencia para saber afrontar cada reto y situación que se presentaron en el camino y hoy poder culminarla con éxito. Suya es toda la gloria y suyos todos mis triunfos.

A mis padres Gerardo Triana y María del Carmen Rubio, porque sin sus enseñanzas, formación, amor y continuo apoyo tal vez no hubiese podido llegar a ser quien soy ahora. Este título más que mío es de ellos porque esto es solo el resultado de lo que ellos han sembrado.

A mis hermanas Carolina Triana y Daniela Triana porque han sido un apoyo fundamental a lo largo de mi vida. Su compañía, confianza en mí y en mis capacidades me han impulsado a seguir adelante en la búsqueda de nuevos logros y el alcance de mis sueños.

A mis demás familiares, amigos, amigas, compañeros y todos los que de una u otra manera han hecho parte de este sueño y han aportado de sí para cada día poder escalar un peldaño más.

AGRADECIMIENTOS

Al Ingeniero Drago Dussich y la Ingeniera Jenny Chaves, Gerente General y Jefe del Departamento de Electrónica de RT Solutions LTDA respectivamente, por su constante apoyo y dirección en el desarrollo de este proyecto.

Al Ingeniero Juan Carlos Mena Moreno, por su dirección para un buen desarrollo del proyecto y alcance de los objetivos.

Al Ingeniero Juan Carlos Ruiz, Director de Tráfico Vial en Imatic Ingeniería Ltda. y su equipo de trabajo por su colaboración y tiempo dedicado a este proyecto.

Al Ingeniero Pablo Ñáñez y demás personal de la Universidad de los Andes por su importante aporte y dedicación.

A mis compañeros de Universidad Jhan Hendry Pineda, Maby Lorena Maldonado, Christian Soto, Paula Alexa Janssen, Julián Ussa y Andrea Franco por sus valiosos aportes tanto a mi vida personal como académica y profesional.

CONTENIDO

	pág.
GLOSARIO	15
RESUMEN	17
INTRODUCCIÓN	18
1. PROBLEMA DE INVESTIGACIÓN	19
1.1. PLANTEAMIENTO DEL PROBLEMA	19
2. JUSTIFICACIÓN	21
3. OBJETIVOS	23
3.1. OBJETIVO GENERAL	23
3.2. OBJETIVOS ESPECÍFICOS	23
4. MARCO DE REFERENCIA	24
4.1. ANTECEDENTES	24
4.1.1. Estrategias de control de tráfico vehicular urbano.	24
4.1.1.1. Control a partir de tiempos fijos.	24
4.1.1.2. Control Adaptativo a partir de buclas magnéticas.	25
4.1.2. Comunicación con la central de tráfico para control adaptativo global.	26
4.1.3. Sistemas de control de tráfico adaptativo existentes en el mercado internacional	26

4.1.3.1. SCOOT.	26
4.1.3.2. SCATS.	28
4.2. MARCO TEÓRICO	29
4.2.1 Programación orientada a objetos.	29
4.2.1.1. Clases y objetos.	30
4.2.1.2. Mensajes y métodos.	30
4.2.1.3. Atributos.	30
4.2.2. Programación orientada a agentes.	31
4.2.3. Teoría evolutiva de juegos.	31
4.2.3.1. Replicator dynamics.	32
4.2.4. Control en lazo abierto vs. Lazo cerrado	33
4.2.4.1. Sistemas de control en lazo abierto.	33
4.2.4.2. Sistemas de control en lazo cerrado	33
4.2.5. Distancia de Hamming	35
4.2.6. Conexión GPRS.	36
4.2.7. Microsimulación de tráfico.	36
4.2.7.1. Software para microsimulación de tráfico VISSIM	38
4.2.7.1.1. El modelo de simulación de tráfico.	39
4.2.8. Sensores Virtuales.	42
4.2.8.1. Sensores de detección de video para carreteras ITERIS.	42
4.2.8.1.1. Sensor de detección de video PICO de ITERIS, INC.	42

4.2.9. Lenguaje de Programación C# (CSharp).	45
5. ANÁLISIS DEL ALGORITMO DE CONTROL DE TRÁFICO MULTINODAL BASADO EN INTELIGENCIA COMPUTACIONAL	47
5.1. ALGORITMO DE CONTROL	47
5.1.1. Modo de funcionamiento del algoritmo.	48
5.1.2. Tiempo total de verde para la red (P).	49
5.1.3. Generación del plan de señales.	50
6. DESARROLLO DE LA APLICACIÓN MOGUT Y CONVERSIÓN DEL ALGORITMO DE CONTROL MULTINODAL AL LENGUAJE DE PROGRAMACIÓN C# (CSHARP)	52
6.1. CLASES DE LA APLICACIÓN EN C#	53
6.1.1. Clase principal.	53
6.1.2. Clase calcular.	53
6.1.3. Clase controlador mallas.	53
6.1.4. Clase intergratorRK4V.	55
6.1.5. Clase configuración.	55
6.2. MEDICION DE LAS COLAS	55
6.3. ALGORITMO DE CONTROL	56
6.4. COMPARACIÓN DE PLANES DE SEÑALES	57
6.5. SELECCIÓN DE PLANES A EJECUTAR	59
6.5.1. Modo egoísta.	59
6.5.2. Modo compartir duración de ciclo.	59

6.5.2.1. Modo Compartir Duración de Ciclo y Reparto de Verde	59
6.5.3. Modos de ponderación.	59
6.5.3.1. Un vehículo un voto.	60
6.5.3.2. Un cruce un voto.	61
6.6. EXTRAPOLACIÓN DEL VALOR DE LA COLA EN ESTADO DE SATURACIÓN	61
6.7. ALMACENAMIENTO Y DEPURACIÓN DE PLANES DE SEÑALES SUGERIDOS	62
6.7.1. Almacenamiento de Planes de Señales Sugeridos	62
6.7.2. Depuración de la base de datos de planes de señales sugeridos.	63
7. VALIDACIÓN DEL ALGORITMO DE CONTROL POR MEDIO DE SIMULACIÓN HARDWARE-IN-THE-LOOP (HIL)	65
7.1. MONTAJE PARA LAS PRUEBAS	65
7.1.1. Descripción de las partes.	65
7.1.1.1. Equipo de control	65
7.1.1.2. MOGUT.	65
7.1.1.3. Central de tráfico IMATIC	66
7.1.1.4. VISSIM 5.40	66
7.1.1.5. Hardware de acople	67
7.1.1.6. Tarjeta de adquisición de datos (NI USB-6509).	69
7.1.2. Descripción del montaje	69
7.2. DISEÑO DE LAS PRUEBAS	70
7.2.1. Prueba de simulación HIL para configuración de tres intersecciones.	71

7.2.2. Prueba de simulación HIL para configuración de 6 intersecciones	72
8. RESULTADOS Y ANÁLISIS	74
8.1. INDICES DE DESEMPEÑO	74
8.1.1. Tiempo de retraso promedio por vehículo (TRPV).	74
8.1.2. Número promedio de paradas por vehículo (NPPV).	74
8.1.3. Distancia total viajada (DTV).	74
8.1.4. Tiempo total de viaje (TTV).	74
8.1.5. Tiempo total de retraso (TTR).	75
8.1.6. Velocidad promedio (VP	75
8.2. RESULTADOS	75
8.2.1. Resultados esperados	75
8.2.2. Resultados obtenidos	75
8.2.2.1. Análisis de resultados para pruebas con red de 3 intersecciones	75
8.2.2.2. Análisis de resultados para pruebas con red de 6 intersecciones.	77
9. INTERFAZ DE INTEGRACIÓN DEL MOGUT CON LA CENTRAL DE TRÁFICO	78
10. CONCLUSIONES	80
BIBLIOGRAFIA	82
ANEXOS	85

LISTA DE CUADROS

	pág.
Cuadro 1. Variables para el solucionador de ecuaciones diferenciales	48
Cuadro 2. Condiciones para determinar tiempo total de verde para la Red	50
Cuadro 3. Características prueba de Simulación HIL para configuración de tres intersecciones	71
Cuadro 4. Características prueba de Simulación HIL para configuración de tres intersecciones	72
Cuadro 5. Resultados de pruebas de simulación HIL con 3 intersecciones	76
Cuadro 6. Resultados pruebas de simulación HIL con 6 intersecciones	77

LISTA DE FIGURAS

	pág.
Figura 1. Funcionamiento sistema SCOOT	27
Figura 2. Sistema de control en lazo abierto	33
Figura 3. Sistema de control en lazo cerrado	34
Figura 4. Distancia de Hamming tomadas para la secuencia correspondiente al símbolo B	36
Figura 5. Diagrama esquemático de un modelo de Microsimulación	37
Figura 6. Comunicación entre el simulador de tráfico y el generador de estados de señales	40
Figura 7. Sensor de detección de video PICO de ITERIS, INC.	44
Figura 8. Especificaciones Técnicas del Sensor PICO de ITERIS, INC.	44
Figura 9. Diagrama de Control en Lazo Cerrado para Control de Tráfico	47
Figura 10. Plan de Señales con dos grupos de señales vehiculares	51
Figura 11. Diagrama de Implementación MOGUT	54
Figura 12. Sensores de detección a) ITERIS b) VISSIM	56
Figura 13. Base de Datos, Tabla Histórico-Colas	57

Figura 14. Diagrama Entidad - Relación	63
Figura 15. Malla para simulación HIL 3 intersecciones – Calle 82 Carreras 51, 51B y 52	67
Figura 16. Malla para simulación HIL 6 intersecciones – Calle 82 y Calle 84 con Carreras 51, 51B y 52	68
Figura 17. Acople de señales 120 VAC a 5 VDC	68
Figura 18. Acople de señales 120 VAC a 5 VDC	69
Figura 19. Diagrama de Interconexión	70
Figura 20. Interfaz de usuario MOGUT – CENTRAL DE TRÁFICO	78

LISTA DE ECUACIONES

	pág.
Ecuación 1. Replicator equation	32
Ecuación 2. Replicator Equation modificada para el caso de estudio	47
Ecuación 3. Sistema matricial de ecuaciones diferenciales para definir reparto de tiempo total de verde	48
Ecuación 4. Sistema matricial de ecuaciones diferenciales para definir reparto de tiempo de verde en una intersección	49
Ecuación 5. Ocupación de un flujo o fila de una intersección	50
Ecuación 6. Ocupación promedio de la Malla	50
Ecuación 7. Cálculo del índice b para grupos de planes de señales ganadores para cada intersección	60
Ecuación 8. Cálculo del valor de ponderación p para el modo "un vehículo un voto"	61
Ecuación 9. Cálculo de extrapolación del valor de la cola	61
Ecuación 10. Criterio para depuración de la base de datos de planes sugeridos	63

LISTA DE ANEXOS

	pág.
Anexo A. Diagrama de clases MOGUT	85
Anexo B. Datasheet Tarjeta de adquisición de datos NI USB-6509	86
Anexo C. Manual de usuario MOGUT	101

GLOSARIO

ALGORITMO: procedimiento de cálculo que consiste en cumplir una serie o conjunto ordenado y finito de instrucciones que conduce, una vez especificados los datos, a la solución que el problema genérico en cuestión tiene para los datos considerados¹.

COLA: cantidad de vehículos en una calle a la espera del derecho a la vía en una intersección.

CRUCE: sinónimo de *intersección*.

DURACION DE CICLO: tiempo que dura una ejecución completa de un plan de señales.

EQUIPO DE CONTROL: dispositivo desarrollado por IMATIC INGENIERIA LTDA., que administra las diferentes funciones y herramientas propias del control de tráfico. Puede gestionar desde una hasta cuatro intersecciones simultáneamente.

GRUPO DE SEÑALES: constituye un semáforo (entendido como el conjunto de luces para el control de tráfico).

HARDWARE-IN-THE-LOOP: es una forma de simulación en tiempo real. Debe incluir la simulación eléctrica de sensores y actuadores. Esta simulación sirve de interfaz entre el modelo de la planta y el sistema integrado bajo testeo.

INTERSECCIÓN: hace referencia a aquellos elementos de la infraestructura vial donde se cruzan dos o más caminos.

¹ VANCELLS, Joan. Algoritmos y Programas. Editorial UOC. p. 7. P01/79008/00053[en línea] [consultado 12 de Enero de 2013]. Disponible en Internet: <http://books.google.com.co/books?id=O55qXjsDjnYC&printsec=frontcover&hl=es>

MALLA: arreglo rectangular de intersecciones viales.

OPTIMIZACIÓN: es el efecto de buscar la mejor manera de realizar una actividad.

PLAN DE SEÑALES: define los intervalos de tiempo de la secuencia de cambios de los grupos de un cruce.

PLAN SUGERIDO: plan no compatible con los cargados en el equipo de control más allá del valor umbral. Quedan latentes en el sistema para ser aprobados por un operador y transmitidos al controlador de tráfico.

SIMULACIÓN: es una técnica numérica para realizar experimentos en una computadora digital. Estos experimentos involucran ciertos tipos de modelos matemáticos y lógicos que describen el comportamiento de sistemas de negocios, económicos, sociales, biológicos, físicos o químicos a través de largos períodos de tiempo.²

² COSS, Raúl. Simulación, Un enfoque práctico. México, Editorial LIMUSA, 2003. 157 p. ISBN 968-18-1506-8. [consultado 12 de Enero de 2013]. Disponible en Internet: http://books.google.com.co/books?id=iY6dI3E0FNUC&printsec=frontcover&hl=es&source=gbs_ge_summary_r&cad=0

RESUMEN

Con el propósito de contribuir al mejoramiento del flujo del tráfico vehicular en una red de tráfico de la ciudad de Barranquilla, el Ingeniero Pablo Ñañez de la Universidad de los Andes diseñó un algoritmo en la Universidad de los Andes de Control de tráfico vehicular basado en inteligencia computacional y posteriormente se realizó la Validación del mismo y su Integración a la Central de Tráfico para su funcionamiento, los cuales se desarrollan a lo largo de este proyecto.

Inicialmente se llevó a cabo un proceso de familiarización y estudio del algoritmo desarrollado por la Universidad de los Andes y el lenguaje de programación C# (C Sharp) en el cual se validaría dicho algoritmo. Para dicho proceso se tuvo en cuenta conceptos de tráfico utilizados tales como planes de señales, duración de ciclo, grupos de señales, estados de los grupos, etc.

Después se realizó la conversión del Algoritmo de Control por etapas al lenguaje C# para su posible integración a la Central de Tráfico, iniciando con la métrica de comparación desarrollada en este algoritmo para determinar la similitud existente entre el plan de señales arrojado por éste y los ya programados en los equipos de control instalados en la red tráfico para el control de cada intersección; continuando con el algoritmo de control central y una posterior integración de estas dos partes.

Para verificar la funcionalidad del algoritmo convertido y la obtención de óptimos resultados, se hicieron pruebas utilizando el software de simulación de tráfico VISSIM en el cual se configuró una red de tráfico con características similares a la red real y también los sensores de conteo de colas en cada intersección los cuales cumplían con la misma función de las cámaras de detección virtual a instalar en la realidad.

Finalmente, se desarrolló la interfaz que integra el algoritmo de control con la Central de Tráfico y sus bases de datos para el monitoreo de eventos, obtención de datos de las colas y transmisión de órdenes de cambio de Plan de Señales a los Equipos de Control por GPRS.

Palabras Claves: control, red urbana de tráfico, mogut, simulación, algoritmo, programación, validación, central.

INTRODUCCIÓN

En áreas urbanas donde los grupos de señales o semáforos que regulan el tránsito vehicular se encuentran a corta distancia (máximo 200 m) unos de otros, la coordinación de dichas señales entre cruces adyacentes es muy importante y provee grandes beneficios a los usuarios de los corredores que hacen parte de la red. Coordinar señales en toda una red urbana de tráfico es mucho más complejo que hacerlo para una sola intersección. Por esta razón este proyecto se desarrolló con el fin de extender la capacidad existente en los equipos de control de tráfico de seleccionar entre los diferentes planes de señales previamente configurados a partir de los conteos vehiculares hechos con sensores de video (iteris); validando y poniendo en marcha un algoritmo que permita combinar esta información con la de cruces adyacentes.

En los capítulos 1 y 2 se presentan el problema al cual busca proponer una posible solución este proyecto y la justificación o razones por las cuales se considera es importante atender esta necesidad y el impacto que generará en el grupo de beneficiados por los resultados que se obtengan, que en este caso son todos los ciudadanos de Barranquilla.

El capítulo 3 comprende los antecedentes del presente proyecto, es decir, aquellos trabajos realizados previamente que permiten o dan pie para la correcta ejecución de éste como las estrategias de control utilizadas actualmente para la regulación del tráfico y el protocolo y medio usado de comunicación entre las estaciones de control en campo y la central de tráfico.

El capítulo 4 expresa claramente el objetivo general del proyecto y las tareas que se realizaron para alcanzarlo.

En el capítulo 5 se describen los diferentes conocimientos teóricos y técnicos que se deben poseer para la comprensión y realización del trabajo planteado en este documento.

Finalmente, los capítulos 6 al 9 describen los pasos realizados para alcanzar el objetivo general y presentan los resultados obtenidos y cómo estos representan un desarrollo exitoso del proyecto.

1. PROBLEMA DE INVESTIGACIÓN

1.1. PLANTEAMIENTO DEL PROBLEMA

El control de tráfico se ha convertido en un tema de interés general para la sociedad y las empresas que trabajan en el campo puesto que la congestión vehicular en las ciudades es cada vez mayor y los controles desarrollados hasta este momento para regular esta situación y el tránsito peatonal no son lo suficientemente óptimos para lograr el desempeño que se desea de manera que tanto los vehículos como los peatones puedan desplazarse dentro de la ciudad de forma rápida para alcanzar sus destinos en la menor cantidad de tiempo posible.

Actualmente, los planes de tiempo que se configuran en los controladores que generan los mandos de cambios de señales en los semáforos son diseñados a partir de estimaciones del comportamiento del tráfico en las diferentes horas del día. Pero, el comportamiento del tráfico en un punto definido de la ciudad es un sistema muy complejo y dinámico por lo cual no se puede asegurar que será el mismo para exactamente el mismo instante de tiempo en dos días diferentes.

Por esa razón, se ha propuesto un proyecto para la ciudad de Barranquilla, el cual se encuentra dividido en dos fases: La primera, es el diseño de un algoritmo para control de tráfico en una malla vehicular (multinodal) basado en la técnica de Inteligencia Computacional conocida como "Replicator Dynamics" utilizando la herramienta de Matlab, a cargo de la empresa IMATIC Ingeniería Ltda., en convenio con la Universidad de los Andes de Bogotá por medio de un acuerdo existente entre IMATIC y la Universidad de los Andes para utilizar el conocimiento y resultados obtenidos en dicha Universidad. En esta etapa se realizaron pruebas con el software para simulación de tráfico VISSIM en la Universidad de los Andes. La segunda fase del proyecto comprende la validación del algoritmo con el desarrollo de pruebas de simulación Hardware-in-the-loop nuevamente con el software VISSIM y utilizando el Equipo para control de tráfico desarrollado por la empresa IMATIC INGENIERIA LTDA., el cual interactúa con el simulador de tráfico, y su implementación en la Central de tráfico donde se contará con información del comportamiento del tráfico en toda la malla vehicular. Es en esta última fase que se centrará el desarrollo de esta pasantía y para lo cual surge la siguiente problemática:

¿Cómo validar e implementar el algoritmo de control de tráfico multinodal diseñado por la Universidad de los Andes en la central de tráfico de Barranquilla, para

extender la capacidad de control que tienen los equipos al flexibilizar la selección entre diferentes diagramas de tiempo previamente configurados en los mismos, con el fin de minimizar el largo de las colas en cada cruce, es decir, la cantidad de vehículos en espera de luz verde, reduciendo el tiempo promedio de espera en el semáforo?

2. JUSTIFICACIÓN

Son muchos los factores sociales que han contribuido a que el tráfico vehicular dentro de las ciudades sea cada vez mayor y, como consecuencia, al incremento de la inseguridad vial tanto para peatones como para los conductores de los vehículos, y de la dificultad para desplazarse de un lugar a otro sin encontrarse envuelto en una situación de congestión vehicular haciendo así el flujo más lento y el alcance de su destino mucho más demorado.

Con la realización de este proyecto se verían beneficiados tanto los usuarios de los diferentes vehículos teniendo en cuenta que al realizar un tipo de control realimentado que utiliza la información recibida de cruces adyacentes, se puede garantizar el flujo continuo y a una velocidad constante durante un tramo de distancia considerable, realizando la menor cantidad de paradas posibles, facilitando así que todas las personas puedan alcanzar sus destinos de una manera más eficiente, rápida y segura.

Además, se realiza una contribución importante desde el punto de vista tecnológico en la utilización de sensores modernos en la detección y conteo de vehículos a partir de la toma de imágenes de video en cada uno de los cruces. Se suma a esto que la conversión de datos de las imágenes tomadas por los sensores se hace localmente en cada una de las estaciones de control y posteriormente se transmite la información vía GPRS (un sistema inalámbrico para la transmisión de datos no conmutada) a la Central de Tráfico.

También, se presenta una alternativa al mejorar la situación de contaminación del aire con el desarrollo del proyecto, puesto que el disminuir el tiempo que tarda un vehículo en llegar a su destino normalmente implica también una reducción en la contaminación que genera dicho vehículo durante el tiempo que deja de estar encendido y circulando en la ciudad. Teniendo en cuenta que el 1% del proceso de combustión del motor de un automóvil se convierte en contaminación y que sería grande el número de vehículos que sufrirían los cambios positivos anteriormente planteados, se puede considerar una contribución importante a la disminución de la contaminación en el Medio Ambiente.

Adicionalmente, para la Universidad Autónoma de Occidente es un aporte significativo en la aplicación de técnicas de Inteligencia Artificial como solución a problemas de control a gran escala, puesto que es un tema que con el paso del tiempo cobra fuerza por ser moderno y una herramienta útil para solucionar problemas complejos y no lineales, los cuales no pueden ser resueltos con técnicas de control tradicional. Es una buena fuente de información para un grupo de investigación que desee basar su trabajo en este campo o simplemente personas de la comunidad educativa que se interesen en el tema.

Finalmente, para RTSolutions Ltda., es también un avance en cuanto al trabajo que han desarrollado en términos de semaforización a lo largo de su historia. Este proyecto presenta nuevas oportunidades para la empresa y abre el panorama de proyectos objetivos a realizar en el futuro. Además, aporta experiencia para la misma en el desarrollo de trabajos basados en inteligencia artificial y manejo de mallas o redes de vías urbanas de transporte para control de tráfico vehicular.

3. OBJETIVOS

3.1. OBJETIVO GENERAL

Validar por medio de simulación el algoritmo de control de tráfico multinodal para extender la capacidad de control que tienen los equipos con el fin de minimizar el largo de las colas en cada cruce reduciendo el tiempo promedio de espera en el semáforo.

3.2. OBJETIVOS ESPECÍFICOS

- Comprender el algoritmo de control multinodal realizado por la Universidad de los Andes en Matlab basado en el modelo de inteligencia computacional Replicator Dynamics.
- Convertir el algoritmo de control multinodal realizado por la Universidad de los Andes en Matlab al lenguaje de programación C# (CSharp).
- Validar el algoritmo por medio de pruebas con ayuda del software de simulación de tráfico VISSIM.
- Realizar la interface para la integración del algoritmo y la información de cada cruce a la Central de tráfico.

4. MARCO DE REFERENCIA

4.1. ANTECEDENTES

4.1.1. Estrategias de control de tráfico vehicular urbano. En el mundo entero, de manera global, existen dos grandes estrategias de control de tráfico urbano comúnmente utilizadas: control a partir de tiempos fijos y control adaptativo a partir de buclas magnéticas. Estas estrategias de control se pueden entender como un control en Lazo Abierto y en Lazo Cerrado respectivamente.

En esta sección se abordan ambas estrategias de control así como sus ventajas y desventajas.

4.1.1.1. Control a partir de tiempos fijos. Es el control comúnmente más utilizado en control de tráfico urbano para el funcionamiento de los semáforos como herramienta principal para la regulación de tráfico. es una estrategia de control del tipo LAZO ABIERTO, la cual se basa en la configuración de planes de señales de tiempo fijo para diferentes rangos horarios del día a partir de estimaciones realizadas del comportamiento del flujo vehicular, resultantes de un estudio preliminar conocido como aforos, en el que se observó dicho comportamiento durante una determinada cantidad de días en diferentes horarios con el fin de predecir cuantos automóviles circulan por ciertas rutas en esos horarios y las posibilidades de formación de congestión vehicular teniendo en cuenta datos como el número de carriles de la vía, circulación en dos o un solo sentido, capacidad vehicular de la vía, etc. Estos planes de tiempo se diseñan e implementan en las estaciones de control.

Ventajas:

- No requiere de interacción continua entre el equipo de control y la central de tráfico.
- No necesita instalación de costosos sensores.
- Es basado en un modelo definido experimentalmente del comportamiento del tráfico.

Desventajas

- Es insensible a las perturbaciones inesperadas en el comportamiento normal del tráfico.

4.1.1.2. Control Adaptativo a partir de buclas magnéticas. Esta una estrategia de control local (para cada intersección individualmente) que utiliza información sensada (lazo cerrado) de las condiciones del tráfico para modificar los tiempos de duración de ciclo, reparto de verdes o desfases de los planes de señales en el equipo de control.

Normalmente, se usan como sensores buclas magnéticas que son instaladas debajo del pavimento, para medir las colas en dicho corredor. Cuando un vehículo pasa por encima del sensor, éste envía un pulso que es interpretado por la estación de control como un ingreso a la cola o una salida de la misma de acuerdo a la ubicación del sensor en el corredor.

Ventajas

- Retroalimentación permanente de las condiciones del tráfico.
- No necesita comunicación permanente con la Central de Tráfico puesto que el control es local.
- Responde a las necesidades de la intersección.

Desventajas

- Altos costos de mantenimiento puesto que se debe levantar el pavimento para reparar los sensores.
- Posibles limitaciones físicas de instalación por el cableado.
- Las vibraciones producidas por los vehículos muy grandes que circulan por el corredor causan con el tiempo que los sensores se empiecen a levantar y a romper la carretera.
- Se puede implementar solamente en intersecciones aisladas porque al ser un control local, su constante variación generaría un desfase con las demás intersecciones.

4.1.2. Comunicación con la central de tráfico para control adaptativo global.

Para realizar un sistema de control adaptativo global, el proceso debe hacerse desde un sitio donde se tenga información de todas las intersecciones de la malla. Puesto que todos los equipos tienen la capacidad de transmitir y recibir información para y de la Central de Tráfico, ésta es la mejor herramienta a utilizar. Los datos sensados en cada intersección son transmitidos desde la estación de control local hacia la Central a la cual se accede para obtener información de toda la malla a urbana a controlar.

La central de tráfico se comunica con los controladores por medio de un protocolo abierto que funciona sobre TCP/IP de forma que la transmisión de datos puede hacerse a través de diversos medios como cable coaxial, fibra óptica, GPRS, WIMAX, etc.

Debido a que la comunicación debe hacerse a grandes distancias las cuales oscilan alrededor de los 22 km, la comunicación se hace por GPRS y utiliza el operador CLARO. Esto presenta las siguientes ventajas con respecto a las demás opciones de sistemas de transmisión de datos:

- Disminución de costos de obra civil al usar transmisión inalámbrica.
- El cliente GPRS puede estar conectado todo el tiempo ya que no se hace uso de los recursos de red mientras no haya transmisión de datos.
- Tarificación por volumen de datos transferidos en lugar de tiempo.
- Excelente velocidad de transmisión.
- Cobertura geográfica muy amplia.³

4.1.3. Sistemas de control de tráfico adaptativo existentes en el mercado internacional

4.1.3.1. SCOOT. Es un Sistema de Control de Tráfico Adaptativo desarrollado en Inglaterra que coordina la operación de de todas las señales de tráfico en un área para dar buen progreso a los vehículos a través de la red de tráfico.

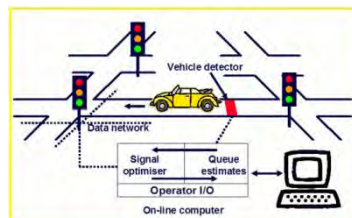
³ DUSSICH, Drago. Comunicaciones inalámbricas para el control del tráfico vehicular. En: Revista Colombiana de Telecomunicaciones. Agosto-Octubre, 2010, vol. 17, no. 57, p. 38-41.

Mientras coordina todas las señales, responde inteligente y continuamente a los cambios del flujo de tráfico y varía durante el día. Remueve la dependencia de sistemas menos sofisticados de planes de señales, para los cuales resulta muy costosa su actualización.

SCOOT detecta vehículos al inicio de una zona inmediatamente anterior a cada intersección controlada. Modela el progreso del tráfico desde el detector hasta la línea de parada, tomando en cuenta el estado de las señales y cualquier cola de vehículos que se pueda presentar como consecuencia.

SCOOT obtiene información sobre los flujos del tráfico desde los detectores. Como sistema adaptativo, SCOOT depende de buena información acerca del tráfico para poder responder a cambios en el flujo. Normalmente se requieren detectores en cada conexión. Su ubicación es importante y normalmente se encuentran en la parte superior del final de cada conexión. Normalmente se usan buclas magnéticas, pero también se encuentran disponibles otros métodos.

Figura 1. Funcionamiento sistema SCOOT



Fuente: How Scoot Works [en línea]. Inglaterra: Peek Traffic Limited, Siemens Traffic Controls y TRL Limited, 2008. Disponible en Internet: <http://www.scoot-utc.com/HowSCOOTWorks.php?menu=Overview>

Cuando los vehículos pasan por el detector, SCOOT recibe la información y convierte los datos en sus unidades internas y los usa para construir "perfiles de flujo cíclicos" para cada una de las conexiones. Los datos de este modelo son usados por SCOOT en tres optimizadores los cuales están continuamente adaptando tres parámetros claves en el control de tráfico – la cantidad de verde para cada conexión, el tiempo entre grupos adyacentes (desfase), y el tiempo permitido para cada conexión en una intersección señalizada (tiempo del ciclo). Estos tres optimizadores son usados continuamente para adaptar dichos

parámetros en todas las intersecciones del área controlada por SCOOT, minimizando el tiempo de verde desperdiciado en las intersecciones y reduciendo las paradas y retrasos al sincronizar grupos de señales adyacentes. Esto significa que los tiempos de las señales evolucionan mientras la situación del tráfico cambia sin ninguna de las interrupciones causadas por los cambios de los planes de tiempo fijos en los sistemas de control de tráfico urbano más tradicionales.⁴

4.1.3.2. SCATS. Es un Sistema de Gestión de Tráfico de Área Amplia que opera bajo el ambiente de Windows. Este sistema controla el ciclo de tiempo, repartos de verde y desfases en las intersecciones de control de tráfico y cruces peatonales. Con la inclusión de detectores de vehículos, puede modificar estos valores para optimizar la operación de forma que se adapte a las condiciones de tráfico dominante. Alternativamente, puede manejar las intersecciones en modo de tiempo fijo en el cual tiene la posibilidad de cambiar los planes por hora del día y día de la semana. Está diseñado para coordinar señales de tráfico para redes o para carreteras principales.

Las conexiones de las intersecciones a un equipo de control de tráfico regional puede ser permanente o de acuerdo a la demanda presente utilizando las opciones de dial-in o dial-out. Cada equipo regional puede gestionar hasta 250 intersecciones. Un sistema SCATS puede tener un máximo de 64 equipos regionales.

La supervisión es proporcionada por una interfaz gráfica de usuario. Hasta 100 usuarios se pueden conectar a un gestor central SCATS al mismo tiempo. Hasta 30 usuarios pueden conectarse a un solo equipo regional simultáneamente. Se encuentran incluidas las opciones de monitoreo del desempeño, notificación de condición de alarma y configuración de los datos. SCATS automáticamente recoge información de alarmas y eventos, datos operacionales y de desempeño y datos históricos. SCATS opera automáticamente, pero se provee intervención de operación en caso de emergencia.

⁴ What is Scoot? [en línea]. Peek Traffic Limited, Siemens Traffic Controls y TRL Limited [consultado 05 de Diciembre de 2011]. Disponible en Internet: <http://www.scoot-utc.com/WhatIsSCOOT.php?menu=Overview>

Este software incluye utilidades para el soporte en la configuración de los equipos, creación de gráficas, producción de reportes del desempeño del tráfico y reportes de incidentes, eventos o alarmas.⁵

4.2. MARCO TEÓRICO

A continuación se presentan los soportes teóricos y conocimientos previos necesarios para el desarrollo de este proyecto como la programación orientada a Objetos y Agentes para la escritura y traducción del algoritmo de control; control en lazo abierto y lazo cerrado, para entender las diferencias y las ventajas que presentan el uno sobre el otro dependiendo de los sistemas a controlar, teniendo en cuenta que en este caso se utiliza el modelo de control en lazo cerrado; la microsimulación de tráfico, puesto que es la herramienta utilizada para realizar pruebas virtuales del funcionamiento del modelo de control planteado; la comunicación GPRS, la cual se utiliza para compartir información entre los controladores instalados en diferentes cruces y la central de tráfico. Finalmente, detalles técnicos y de funcionamiento de los sensores de vídeo ITERIS a utilizar para el conteo de vehículos en la implementación final de la aplicación de control.

4.2.1 Programación orientada a objetos. La Programación Orientada a Objetos (POO) es un modelo de programación que utiliza objetos, ligados mediante mensajes, para la solución de problemas. La idea central es simple: organizar los programas a imagen y semejanza de la organización de los objetos en el mundo real.

En un determinado problema que se desea resolver se pueden identificar entidades de interés, las cuales pueden ser objetos potenciales que poseen un conjunto de *propiedades* o atributos, y un conjunto de *métodos* mediante los cuales muestran su comportamiento. Además, se pueden ver un conjunto de interrelaciones entre ellos conducidas por mensajes a los que responden mediante métodos. Estas entidades, para el fin de la programación, se pueden considerar como **objetos**.

⁵ SCATS [en línea]. Australia: Transport, roads and maritime services, NSW Government [consultado 05 de Diciembre de 2011]. Disponible en Internet: http://www.scats.com.au/product_base_packg_compnts.html

Así desde el punto de vista de la POO, un programa es un conjunto de objetos que dialogan entre sí para realizar las distintas tareas programadas. A continuación se describen las partes de un programa orientado a objetos⁶:

La POO se basa en la observación de que, en el mundo real, los objetos se construyen a partir de otros objetos. La combinación de estos objetos es un aspecto de dicha programación, pero también incluye mecanismos y características que hacen que la creación y el uso de objetos sea sencillo y flexible. Un mecanismo importante es la *clase*, y el encapsulamiento y la herencia son dos propiedades o características poderosas.

4.2.1.1. Clases y objetos. Una **clase** es la generalización de un tipo específico de objetos, pero cada objeto que se construya de esa clase tiene sus propios datos. Un **objeto** de una clase se crea cuando se define una variable de dicha clase. Cuando se escribe un programa en un lenguaje orientado a objetos no se definen objetos sino clases de objetos, las cuales se convierten en el modelo para muchos objetos con características similares.

4.2.1.2. Mensajes y métodos. Los objetos son entidades que tienen propiedades y características particulares conocidas como *atributos*, y la forma de operar sobre ellos se conoce como **método**.

En la ejecución de un programa orientado a objetos, estos responden todo el tiempo a interacciones entre ellos mismos. Este tipo de comunicación entre objetos se conoce como **mensajes**.

Un método en una clase define cual debe ser la acción a tomar por parte de un objeto como respuesta al mensaje asociado con dicho método. Pero también, un método puede enviar mensajes a otros objetos solicitando algún tipo de información.

4.2.1.3. Atributos. Los **atributos** son las características que diferencian un objeto de otro. Generalmente, los atributos de un objeto de una clase se ocultan de los

⁶ CEBALLOS, Francisco J. Microsoft C# Curso de Programación. México.:Alfaomega Grupo Editor, 2007. p. 33-38. ISBN 978-970-15-1222-7.

usuarios del mismo con el fin de que solo se pueda acceder a ellos utilizando los métodos que implementa la clase para tal fin.

4.2.2. Programación orientada a agentes. La Programación Orientada a Agentes es una nueva estructura computacional que puede ser vista como una especialización de la Programación Orientada a Objetos. A diferencia de esta última donde los objetos tienen muchos métodos con parámetros de variables asignados a ellos, comúnmente los agentes tienen un solo método con un solo parámetro. Este parámetro es un tipo de mensaje que es interpretado por el agente receptor de una forma específica para dicho agente o clase de agentes.

En Inteligencia Artificial, cuando se utiliza el término "agente", se refiere a una entidad que funciona continua y autónomamente en un ambiente en el cual toman lugar otros procesos y existen otros agentes. El sentido de autonomía no es preciso, pero se utiliza el término para expresar que las actividades de los agentes no requieren intervención humana constante.

El estado de un agente consiste en componentes tales como creencias, decisiones, capacidades y obligaciones; por esta razón el estado de un agente es conocido como su *estado mental*. Cualquier hardware o software se puede considerar un agente si opera bajo estos términos mentales.⁷

4.2.3. Teoría evolutiva de juegos. La teoría evolutiva de juegos trata dinámicas de juego evolutivas y determinísticas las cuales describen la manera como las frecuencias de estrategias dentro de una población cambian en el tiempo, de acuerdo al éxito de las estrategias. La dinámica evolutiva de juegos es la aplicación de métodos dinámicos poblacionales a la teoría de juegos. Esto requiere ciertas modificaciones en el acercamiento conceptual básico. A riesgo de sobre-simplificación, se puede decir que la teoría de juegos clásica trata con un individuo racional, o "jugador", el cual se dedica a una determinada interacción o "juego" con otros jugadores y tiene que decidir entre diferentes opciones, o "estrategias", con el fin de maximizar una "rentabilidad" que depende de las

⁷ SHOHAM, Yoav. Agent-oriented Programming. En: Artificial Intelligence 60. Stanford: Elsevier, 1993. p. 51-52. [consultado 24 de Noviembre de 2011]. Disponible en Internet: <http://123seminaronly.com/Seminar-Reports/050/62112045-Agent-Oriented-Programming.pdf>

estrategias de los demás jugadores (los cuales, a su vez, intentan maximizar su rentabilidad). Por el contrario, la teoría de de juegos evolutiva trata con poblaciones enteras de jugadores, todos programados para usar alguna estrategia (o tipo de comportamiento). Las estrategias con una alta rentabilidad se extenderán dentro de la población (esto puede alcanzarse aprendiendo, copiando o heredando estrategias, o incluso por infección). Las rentabilidades dependen de las acciones de los demás jugadores y, por lo tanto, en las frecuencias de las estrategias dentro de la población. Puesto que estas frecuencias cambian de acuerdo a las rentabilidades, se produce un bucle de retroalimentación. La dinámica de este bucle de retroalimentación es el objeto de estudio de la teoría de juegos evolutiva. Uno de los modelos más destacados que hace parte de la teoría evolutiva de juegos es el de Replicator Dynamics.

4.2.3.1. Replicator dynamics. Consideremos una población que consiste de n tipos, y que x_i sea la frecuencia de tipo i . Luego, el estado de la población está dado por $x \in \mathcal{S}_n$ (conjunto de n estrategias. Ahora debemos asumir que todos los x_i son funciones diferenciables en el tiempo t y postular una ley de movimiento para $x(t)$. Si los individuos se encuentran al azar y luego participan en un juego simétrico con función de rentabilidad f , luego $f(x)_i$ es la rentabilidad esperada para un individuo de tipo i y $\bar{f}(x)$ es la rentabilidad promedio de la población en el estado x . Asumamos que la tasa de crecimiento per capita, es decir, la derivada logarítmica $(\log x_i)' = \dot{x}_i/x_i$, está dada por la diferencia entre la rentabilidad para el tipo i y la rentabilidad promedio de la población. Esto produce la conocida Replicator Equation.⁸

Ecuación 1. Replicator equation

$$\dot{x}_i = x_i((f(x))_i - \bar{f}(x))$$

Para $i = 1, \dots, n$ y $f(x)_i$ definida por el usuario de acuerdo al caso de estudio.

⁸ HOFBAUER, Josef. Evolutionary Game Dynamics. En: Bulletin of the American Mathematical Society [en línea], Vol. 40, Number 4. Electronically published, 2003. p. 479-482. [consultado 21 de Junio de 2012]. Disponible en Internet: <http://homepage.univie.ac.at/josef.hofbauer/03bams.pdf>

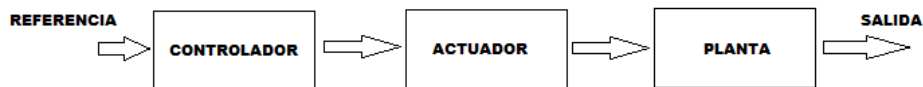
4.2.4. Control en lazo abierto vs. Lazo cerrado⁹

4.2.4.1. Sistemas de control en lazo abierto. Son aquellos sistemas en los cuales la salida no tiene efecto sobre la acción de control, es decir, no se mide la salida ni se realimenta para hacer una comparación con la entrada.

En cualquier sistema de control en lazo abierto, la salida no se compara con la entrada de referencia. Así, a cada entrada de referencia le corresponde una condición de operación fija; como resultado de ello, la precisión del sistema depende de la calibración. Puesto que ante la presencia de perturbaciones el sistema de control en lazo abierto no realiza la tarea deseada, en la práctica, solo se usa este tipo de control si no hay perturbaciones internas ni externas.

Una característica importante para reconocer que un sistema de control está en lazo abierto es que opere con una base de tiempo, como por ejemplo, el sistema de tradicional utilizado en el control de tráfico.

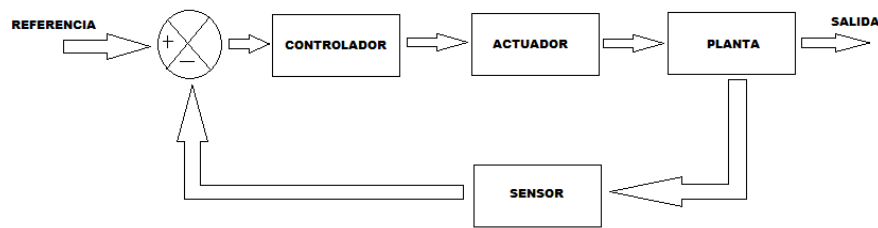
Figura 2. Sistema de control en lazo abierto



4.2.4.2. Sistemas de control en lazo cerrado. Son aquellos sistemas de control realimentados. En este tipo de sistemas, se alimenta el controlador con la señal de error de actuación que es la diferencia entre la entrada de referencia del sistema y la señal de realimentación, con el fin de reducir el error y llevar la salida a un nivel o valor deseado.

⁹ OGATA, Katsuhiko. Ingeniería de Control Moderna. 4 ed. Madrid: Pearson Educación, S.A., 2003. p. 6-7. ISBN 84-205-3678-4.

Figura 3. Sistema de control en lazo cerrado



Una ventaja importante de los sistemas de control en lazo cerrado es que el uso de la realimentación hace que el sistema sea casi insensible a cualquier perturbación externa que se pueda presentar o a cambios en los parámetros internos del sistema. Además es posible realizar un control adecuado de este tipo, para una planta determinada, con componentes poco precisos y por lo tanto baratos.

Pero, desde el punto de vista de la estabilidad, un sistema de control en lazo abierto es mucho más fácil de desarrollar ya que éste no es un aspecto tan importante en estos casos, mientras que en el lazo cerrado, es más complicado e importante obtener la estabilidad de un sistema porque pequeñas variaciones en los parámetros internos pueden causar la presencia de grandes oscilaciones en la respuesta del sistema. Además, al utilizarse más componentes para la construcción del lazo cerrado, la potencia consumida por el sistema es mayor, por lo cual su costo podría ser muy grande.

Por lo tanto, si se desea disminuir la potencia requerida por un determinado sistema o se requiere alta estabilidad, es más recomendable utilizar un sistema de control en lazo abierto siempre y cuando sea posible implementarlo; pero, si lo que se requiere es inmunidad del sistema en medio de perturbaciones tanto internas como externas, lo mejor es implementar un sistema de control en lazo cerrado.

La estrategia de control propuesta en los Antecedentes de este documento plantea un sistema de Control en Lazo cerrado, puesto que existe una realimentación del sistema reflejada en la medición de las colas utilizando sensores de video, y su procesamiento para la aplicación de una nueva acción de control.

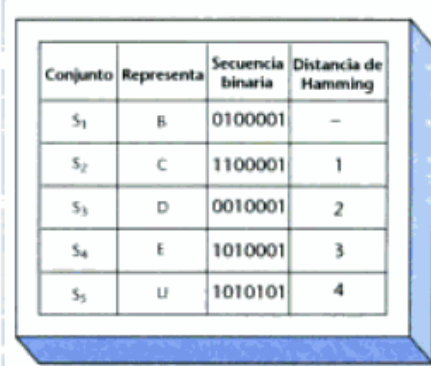
4.2.5. Distancia de Hamming.¹⁰ La distancia de Hamming se define como el número de bits en que difieren dos secuencias binarias, S1 y S2, de la misma longitud.

Sea, por ejemplo, la secuencia S1, que corresponde a la letra B en el Alfabeto Internacional N° 5 de la UIT-T, y sean también las secuencias S2, S3, S4 y S5, correspondientes a otras letras del mismo alfabeto. Si comparamos bit a bit de igual peso, la primera de las secuencias con las restantes, la distancia de Hamming estará dada por la cantidad de bits en que difieren uno a uno, los respectivos conjuntos. Los resultados de este ejemplo se muestran en la figura 4. A efectos de la comparación y del cálculo de la distancia de Hamming, se ha tomado como referencia la secuencia S1. Como se puede observar, en este ejemplo la distancia de Hamming varía de 1 a 4.

La distancia de Hamming está relacionada con la probabilidad de error. En efecto, cuanto mayor sea la distancia mínima entre los símbolos de un código dado, menor será la probabilidad de cometer errores. Sin embargo, aumentar la distancia de Hamming mínima entre símbolos significa codificar menos símbolos con igual número de bits. En otras palabras, aumentando la redundancia se logra disminuir la probabilidad de cometer errores.

¹⁰ CASTRO, Antonio y FUSARIO, Ruben. Teleinformática: para ingenieros en sistemas de información. 2 ed. España: Editorial REVERTÉ, S. A., 1999. 396 p. ISBN-84-291-4391. [consultado 17 de Mayo de 2012]. Disponible en Internet: http://books.google.com.co/books?id=ZOZp72uLGMwC&printsec=frontcover&hl=es&source=gbs_ge_summary_r&cad=0

Figura 4. Distancia de Hamming tomadas para la secuencia correspondiente al símbolo B



Conjunto	Representa	Secuencia binaria	Distancia de Hamming
S ₁	B	0100001	–
S ₂	C	1100001	1
S ₃	D	0010001	2
S ₄	E	1010001	3
S ₅	U	1010101	4

Fuente: CASTRO, Antonio y FUSARIO, Ruben. Teleinformática: para ingenieros en sistemas de información. 2 ed. España: Editorial REVERTÉ, S. A., p. 419. Disponible en Internet:

<http://books.google.com.co/books?id=ZOZp72uLGMwC&pg=PA419&dq=distancia+hamming&hl=es&sa=X&ei=yFI3Ue-clIno9ASCjYHAAQ&ved=0CC4Q6AEwAA#v=onepage&q=distancia%20hamming&f=false>

4.2.6. Conexión GPRS. General Packet Radio Service es una extensión del sistema GSM, para la transmisión de datos no conmutada o por paquetes. Con GPRS se pueden utilizar servicios como Wireless Application Protocol (WAP), servicio SMS, MMS, Internet y para servicios de comunicación como el correo electrónico y la World Wide Web.

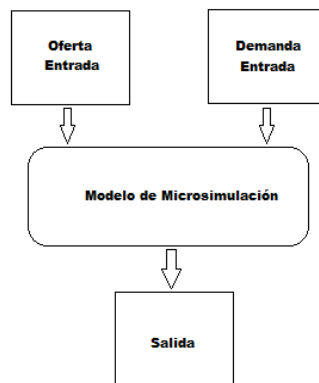
La transferencia de datos por GPRS se cobra por unidad de volumen transmitida y no por tiempo como lo hacen los sistemas tradicionales de conmutación. Esto le da un valor agregado importante puesto que no se cobra mientras el usuario se encuentre en estado de inactividad.

4.2.7. Microsimulación de tráfico. Los modelos de microsimulación de tráfico buscan representar la interacción de los sistemas típicos (por ejemplo, la oferta: calles, intersecciones, control de tráfico, etc.) y los usuarios (por ejemplo, la demanda: rutas, características del conductor, etc.). Se refiere a estos modelos

como micro porque operan a un nivel individual (por ejemplo, los vehículos, las personas). Se refiere a ellos como simulación porque buscan modelar los procesos internos del sistema (por ejemplo, las decisiones de los conductores, características del vehículo, operaciones de señales de tráfico, etc.) y no simplemente la salida del sistema. Estos modelos de microsimulación son populares porque los sistemas que ellos representan son tan complejos que los modelos macroscópicos son insuficientes para hacerlo.

La mayoría de los desarrolladores de modelos de transporte consideran más útil tratar el componente físico de la oferta como una representación matemática de nodos y conexiones, cada una con sus propios atributos. En esencia, las conexiones unen los nodos y por consiguiente, tienen una dirección asociada a ellos, es decir, de un nodo "a" a un nodo "b". Se refiere a la red entera como un gráfico dirigido.¹¹

Figura 5. Diagrama esquemático de un modelo de Microsimulación



Fuente: SPIEGELMAN, Clifford H. PARK, Eun S. RILETT, Laurence R. Transportation: Statistics and Microsimulation. Boca Ratón, FL.: Chapman & Hall/CRC Press, 2011. 355 p. ISBN 978-1-4398-0023-2.

¹¹ SPIEGELMAN, Clifford H. PARK, Eun S. RILETT, Laurence R. Transportation Statistics and Microsimulation. Boca Ratón, FL.: Chapman & Hall/CRC Press, 2011. 355 p. ISBN 978-1-4398-0023-2.

4.2.7.1. Software para microsimulación de tráfico VISSIM¹². VISSIM es un software de simulación de tráfico a nivel microscópico, transporte público y peatones. Este software pertenece a la PTV Vision® suite. Es una de las herramientas más poderosas disponible para la simulación del flujo de tráfico multimodal, incluyendo automóviles, camiones, autobuses, pesados trenes, tranvías, metro ligero, motocicletas, bicicletas y peatones. A través de la implementación de multihilos, VISSIM funciona de manera óptima en computadoras con multiprocesador y multinúcleo.

VISSIM les permite a sus usuarios encontrar una solución que toma en cuenta tanto la calidad del tráfico y del transporte como la seguridad y los costos que dicha solución implica.

El programa puede analizar operaciones de transporte público y privado bajo ciertas condiciones tales como: configuración de carriles, composición de vehículo, señales de tráfico, paradas de transporte público, etc., lo que la convierte en una herramienta útil para la evaluación de varias alternativas basadas en ingeniería de transporte y la planeación de medidas de efectividad. Por consiguiente, los flujos de peatones también pueden ser modelados, ya sea de manera exclusiva o combinados con tráfico privado y/o transporte público.

VISSIM puede ser aplicada como una herramienta útil en una variedad de ajustes en problemas de transporte. La siguiente lista provee una selectiva muestra de aplicaciones previas de VISSIM:

- Desarrollo, evaluación y calibración de lógicas de prioridad de señales: VISSIM puede usar varios tipos de lógica de control de señales. Además de la funcionalidad incorporada de tiempos fijos, hay varios sistemas de control de señales por sensado de vehículos, idénticos a los paquetes de software de control de señales instalados en el campo. En VISSIM algunos de ellos están incorporados, otros pueden ser acoplados utilizando componentes adicionales y otros más pueden ser simulados a través de generadores de estados de señales externos (VAP) que permiten el diseño de una lógica de control de señales definida por el usuario. Así, todos los controles de señales pueden ser modelados y simulados virtualmente dentro de VISSIM ya sea que los detalles

¹² PTV AG. VISSIM 5.40 – User Manual. Alemania, 2012.1 archivo de computador.

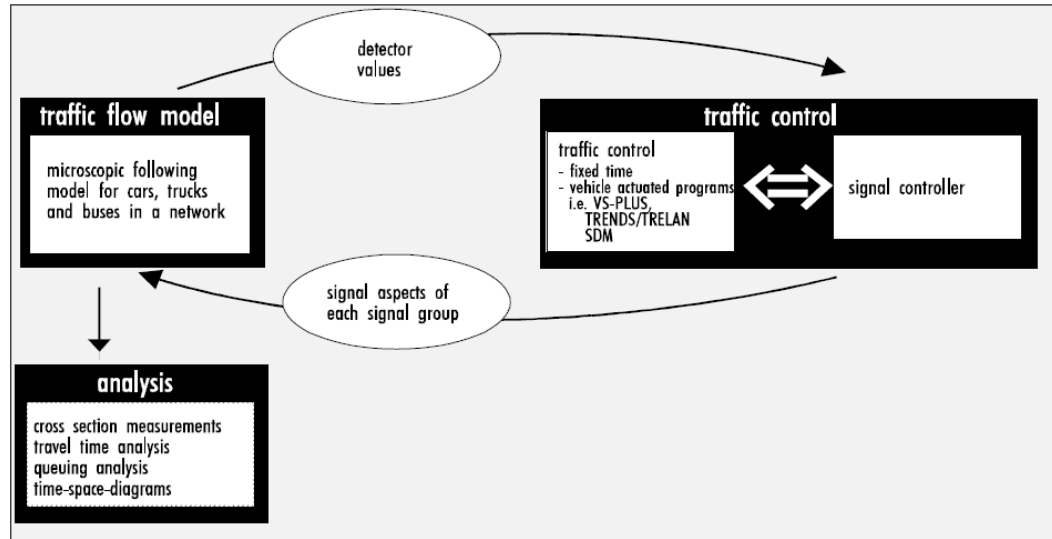
del controlador se encuentren disponibles o que haya una interfaz directamente de VISSIM.

- Evaluación y optimización de operaciones de tráfico en una red combinada de señales de tráfico actuadas y coordinadas.
- Estudios de factibilidad e impacto del tráfico al integrar el tren ligero a redes de calles urbanas.
- Análisis de áreas de baja velocidad.
- Fácil comparación de alternativas de diseño incluyendo intersecciones señalizadas y controladas por señales de detención, rotondas.
- Análisis de capacidad y operaciones de complejos diseños de estaciones para el tren ligero y sistemas de buses han sido analizados con VISSIM.
- Soluciones de tratamiento preferencial para buses (por ejemplo, saltos de colas, carriles solo para buses) han sido evaluadas con VISSIM.
- Con su modelo incorporado de Tareas Dinámicas, VISSIM puede responder preguntas sobre selección de rutas, como los impactos de las señales de mensajes variables o la posibilidad de desviación del tráfico a los barrios de redes con tamaños de hasta una ciudad mediana.
- El modelado y simulación de flujos de peatones – en calles y edificios – permiten una amplia gama de nuevas aplicaciones. VISSIM puede también simular y visualizar las interacciones entre el tráfico vehicular y los peatones.

4.2.7.1.1. El modelo de simulación de tráfico. El paquete de simulación VISSIM consiste internamente de dos partes diferentes, intercambiando llamadas de detectores y el estado de señales a través de una interfaz. La simulación genera una visualización en línea de operaciones de tráfico y, fuera de línea, la generación de archivos de salida que recopilan información estadística como los tiempos de viaje y longitud de las colas.

El **simulador de tráfico** es un modelo de simulación microscópica del flujo del tráfico que incluye el movimiento de los carros y la lógica de cambio de carril. El **generador de estado de señales** es un software de control de señales que lee la información de los detectores (sensores) desde el simulador de tráfico en una base de tiempo discreta (hasta un mínimo de 1/10 seg). Luego, éste determina el estado de las señales para el siguiente instante de tiempo y devuelve esta información al simulador de tráfico.

Figura 6. Comunicación entre el simulador de tráfico y el generador de estados de señales



Fuente: PTV AG. VISSIM 5.40 – User Manual. Alemania, 2012.1 archivo de computador.

La precisión del modelo de simulación de tráfico depende principalmente de la calidad del modelado de los vehículos, por ejemplo, la metodología de movimiento de los vehículos a través de la red. En contraste con modelos menos complejos que usan velocidades constantes y lógica de movimiento de los carros determinística, VISSIM usa el modelo del comportamiento psico-físico del conductor desarrollado por WIEDEMANN (1974). El concepto básico de este modelo es que el conductor de un vehículo moviéndose más rápido comienza a desacelerar al alcanzar su umbral de percepción individual de un vehículo que se mueve más lento. Puesto que no puede determinar exactamente la velocidad de dicho, su velocidad caerá debajo de la misma hasta que empiece a acelerar ligeramente de nuevo después de alcanzar otro umbral de percepción. Finalmente, resulta en un proceso iterativo de aceleración y desaceleración.

Las distribuciones estocásticas de velocidad y el separado de umbrales replican las características del comportamiento de un conductor. El modelo ha sido calibrado por medio de múltiples mediciones de campo en The Technical University of Karlsruhe (desde 2009 KIT – Karlsruher Institut für Technologie), Alemania. Las mediciones periódicas de campo y las actualizaciones resultantes

de los parámetros del modelo aseguran que son tenidos en cuenta los cambios en el comportamiento del conductor y las mejoras de los vehículos.

El simulador de tráfico de VISSIM no solo les permite a los conductores en autopistas de varios carriles reaccionar ante vehículos que los preceden (4 por defecto), sino también ante vehículos vecinos en los carriles adyacentes. Además, al acercarse a las señales de tráfico, resulta un estado de alerta más alto para los conductores que se encuentran a una distancia de 100 metros de la línea de parada.

VISSIM simula el flujo del tráfico moviendo "unidades de vehículo-conductor" a través de la red. Cada conductor con sus características de comportamiento específico es asignado a un vehículo específico. Como consecuencia, el comportamiento del conductor corresponde a las capacidades técnicas de su vehículo. Los atributos que caracterizan a cada unidad vehículo-conductor pueden ser divididos en tres categorías:

- Especificaciones técnicas del vehículo, como por ejemplo:
 - Longitud.
 - Velocidad máxima.
 - Aceleración potencial.
 - Posición actual en la red.
 - Velocidad y aceleración actuales.
- Comportamiento de las unidades vehículo-conductor, como por ejemplo:
 - Umbrales de sensibilidad psico-física del conductor (habilidad para estimar, agresividad).
 - Memoria del conductor.
 - Aceleración basada en la velocidad actual y la velocidad deseada por el conductor.
- Interdependencia de las unidades vehículo-conductor, como por ejemplo:
 - Referencia a liderar y seguir vehículos en su propio carril y en los adyacentes.
 - Referencia al corredor actual y la siguiente intersección.
 - Referencia a la siguiente señal de tráfico.

4.2.8. Sensores Virtuales. Los sensores virtuales, mejor conocidos como "Observadores" o "Estimadores de Estado", son programas de computadora que a partir de las mediciones obtenidas por sensores reales realiza una estimación de las variables de proceso que no cuentan con un sensor real asociado a ellas.

4.2.8.1. Sensores de detección de video para carreteras ITERIS. Iteris utiliza la aplicación de video en una amplia gama de aplicaciones en gestión de tráfico, incluyendo sistemas tanto de control de intersecciones como de gestión de autopistas. Su avanzada tecnología en procesamiento de imágenes es usada en sus productos para detectar la presencia de vehículos, contar, medir velocidad y otros datos de tráfico utilizados en los sistemas de control de tráfico. Esto le da a quienes trabajan en el control de tráfico un kit de herramientas poderoso para mitigar la congestión en las carreteras modificando el tiempo de las señales para el tráfico o detectando incidentes rápidamente.¹³

Los beneficios que brinda la Detección de video en contraste a los sensores inductivos ubicados bajo tierra son:

- La detección de video es muy flexible en cuanto a la ubicación en la zona de detección puesto que no es fijo como lo son los sensores inductivos.
- La fiabilidad de los sensores inductivos se basa en gran manera a las condiciones en las que se encuentra la superficie de la carretera. Esta superficie se ve impactada por el paso de camiones pesados, temperaturas extremas y el mantenimiento que recibe.
- La detección de video no se ve afectada por la repavimentación vial.
- La detección de video es rentable con costos iniciales comparables a los de los bucles de inducción y costos de ciclo de vida muy superiores.
- La detección de video provee beneficios adicionales a quien trabaja con tráfico: las imágenes de video para vigilancia, verificación de incidentes y seguridad.

4.2.8.1.1. Sensor de detección de video PICO de ITERIS, INC. Pico, es un pequeño sistema de detección de huellas en video adaptado a una gran variedad de necesidades en la regulación de tráfico. Está específicamente diseñado para mercados mundiales. Contiene el beneficio adicional de video en vivo, lo cual hace

¹³ Iteris, Inc. Roadway Sensors. 2010. [consultado 05 de Diciembre de 2011]. Disponible en Internet: <http://www.iteris.com/vvd.aspx>

la instalación, configuración y verificación de zona mucho más fácil. Pico es el producto perfecto para aplicaciones pequeñas a medianas en intersecciones que requieren hasta ocho zonas de detección.

El sistema integrado provee el sensor de detección de video en un paquete robusto y discreto, reduciendo las posibilidades de robo y vandalismo. El sensor se puede montar a una baja altura, lo cual elimina la necesidad de postes adicionales o costosos accesorios de montaje. El sistema PICO incluye un sensor de cámara de baja potencia, bajo voltaje y resistente a la intemperie, con un procesador de detección incorporado y un controlador de interfaz que reside en el gabinete de tráfico.

El diseño robusto del compacto y ligero sensor de cámara PICO hace que éste sea tolerante a diferentes alturas de montaje en la intersección, proporcionando una configuración rápida para los técnicos. El controlador de interfaz está compuesto por la Cabina de Comunicación Pico (PCC), módulo que soporta hasta cuatro cámaras y dispone de 16 salidas de colector abierto para el controlador de señales de tráfico para su uso como indicadores de presencia de vehículos. Esto permite que se ejecute una acción de señal optimizada para el eficiente flujo del tráfico.

PICO también está diseñado para reducir el tiempo necesario para la configuración de las zonas de detección de vehículos. Las zonas virtuales de detección se pueden determinar ya sea con una computadora portátil convencional con la herramienta de configuración de PICO, o simplemente con un mouse y un monitor de video. La herramienta de configuración permite que los ajustes realizados se guarden offline para facilitar el reemplazo rápido de la cámara sin necesidad de reiniciar las zonas, así como proporcionar las actualizaciones de firmware. La interfaz amigable con el usuario les permite a los técnicos de campo configurar y verificar las zonas de detección en solo unos pocos minutos con el mínimo de entrenamiento.

PICO utiliza los algoritmos de ITERIS de detección de video probados en campo los cuales operan en más de 80 mil sistemas de cámara desplegados en todo el mundo. La configuración de los sensores de detección de video requiere mucho menos mano de obra que los sensores convencionales bajo tierra y es más seguro para los instaladores puesto que no están en medio de la carretera durante la

mayor parte del tiempo de instalación. La detección de video es una inversión de alto rendimiento que se puede expandir para satisfacer las crecientes necesidades tales como el conteo de vehículos, bicicletas y detección de incidentes.¹⁴

Figura 7. Sensor de detección de video PICO de ITERIS, INC.



Fuente: Iteris, Inc. Pico: Compacto, easy-to-use video detection for optimised signal actuation. 2011, p. 1 [consultado 05 de Diciembre de 2011]. Disponible en Internet: http://www.iteris.com/upload/datasheets/Pico_web.pdf

Figura 8. Especificaciones Técnicas del Sensor PICO de ITERIS, INC.

	SENSOR	INTERFACE CONTROLLER
Features	6mm diagonal colour imager	DIN rail mount
	>400 TVL	Compact design
	<0.5 lux	Easy terminations
	>46 dB S/N ratio	16 fail safe open-collector
	3mm or 8mm fixed focal length lens	outputs (30 VDC @ 50 ma max.)
Mechanical		
Enclosure	Molded polycarbonate housing,	n/a
	IP67 rating	
Size	203.2mm x 101.6mm x 101.6mm	76.2mm x 152.4mm x 63.5mm
Weight	0.68 kg	0.2 kg
Environmental		
Operating Temperature	-34°C to +74°C	-34°C to +74°C
Humidity	100% relative	90% relative, non-condensing
Vibration	0.5G, 3 axes, 5-30 Hz	0.5G, 3 axes, 5-30 Hz
Shock	10G, 3 axes	10G, 3 axes
Power	24 VDC, 0.5 amps	12 VDC, 0.3 amps
	12 watts	3.6 watts

Fuente: Ibíd. p.2.

¹⁴ Iteris, Inc. Pico: Compacto, easy-to-use video detection for optimised signal actuation. 2011. [consultado 05 de Diciembre de 2011]. Disponible en Internet: http://www.iteris.com/upload/datasheets/Pico_web.pdf

Esta herramienta permite también administrar los corredores que se manejan en el sistema. Dentro de esa configuración permite: asignar un nombre al corredor; un color, que permite diferenciar el corredor al visualizarlo en el mapa; la ubicación del corredor, definido por el Departamento y la ciudad; visibilidad del corredor, donde se define si se desea visualizarlo en el mapa o no; y la velocidad máxima permitida en el corredor en Km/h.

Además, la Central permite gestionar también la herramienta de mensajes de texto donde se definen los destinatarios de los mensajes SMS para el envío de los eventos transmitidos por los equipos; se ingresa el nombre de la persona y su número celular que puede ser de cualquier operador disponible en el país.

En la sección de Reportes, se consultan y analizan los reportes detallados de los eventos recibidos en la central; se pueden generar por fecha, por equipo o por cruce.

También posee una sección llamada Mapa, en la que se controla, monitorea y configura los objetos o componentes de la Central de Tráfico, en un mapa geo-referenciado a la ciudad controlada, utilizando la herramienta desarrollada por GOOGLE conocida como Google Maps.

Adicionalmente, en la opción de Video Wall se ajustan los parámetros que el Video Wall manejará por defecto como el período de actualización y el nivel de zoom, donde se deben configurar las opciones de paso, offset X y Y, visibilidad de objetos en el mapa y Centro.

Finalmente, en el menú Administrar Usuarios se crean, editan y administran los usuarios que tienen acceso a la Central de Tráfico.

4.2.9. Lenguaje de Programación C# (CSharp). C# es un lenguaje orientado a objetos elegante y con seguridad de tipos que permite a los desarrolladores compilar diversas aplicaciones sólidas y seguras que se ejecutan en .NET Framework. Se puede utilizar C# para crear aplicaciones cliente de Windows tradicionales, servicios Web XML, componentes distribuidos, aplicaciones cliente-servidor, aplicaciones de base de datos, y mucho más. Visual C# 2010 proporciona un editor de código avanzado, cómodos diseñadores de interfaz de usuario, depurador integrado y numerosas herramientas más para facilitar el

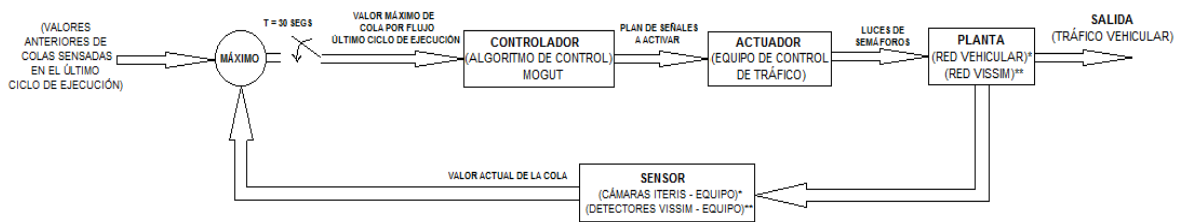
desarrollo de aplicaciones basadas en la versión 4.0 del lenguaje C# y la versión 4 de .NET Framework.¹⁵

¹⁵ <http://msdn.microsoft.com/library/z1zx9t92>

5. ANÁLISIS DEL ALGORITMO DE CONTROL DE TRÁFICO MULTINODAL BASADO EN INTELIGENCIA COMPUTACIONAL

En este capítulo se explicará y describirá el algoritmo de control basado en el modelo de Replicator Dynamics y que ha sido desarrollado por el Ingeniero Pablo Ñañez Ojeda de la Universidad de los Andes en convenio con IMATIC INGENIERIA LTDA y que se puede identificar en la Figura 9.

Figura 9. Diagrama de Control en Lazo Cerrado para Control de Tráfico



* Se utiliza en la realidad.

** Se utiliza en simulación.

5.1. ALGORITMO DE CONTROL

El algoritmo desarrollado para el procesamiento de la información y generación de un nuevo plan de señales está basado en un modelo conocido como *Replicator Dynamics*, una dinámica de juegos usada en la Teoría Evolutiva de Juegos (Ver sección 4.2.3.1). El modelo viene dado por la ecuación 1.

Para el caso en estudio las variables representan lo descrito en el cuadro 1. Reemplazando estas condiciones en la Ecuación 1 obtenemos:

Ecuación 2. Replicator Equation modificada para el caso de estudio

$$\dot{x}_i = -\frac{a}{p}x_i + a_i$$

5.1.1. Modo de funcionamiento del algoritmo. En el algoritmo se identifican dos niveles de jerarquía. El nivel de jerarquía mayor define qué cantidad de tiempo de verde del total asignado a la Red le corresponde a cada cruce. Para ello obtenemos un sistema de Ecuaciones Diferenciales Ordinarias o de Primer Orden, que parte del modelo descrito en esta sección, de la forma $EX' = AX + B$ y representado matricialmente en la Ecuación 3.

Cuadro 1. Variables para el solucionador de ecuaciones diferenciales

Variable	En toda la ted	En cada intersección
i	Índice de cada intersección	Índice de cada flujo
x_i	Tiempo de verde para cada intersección	Tiempo de verde para cada flujo
$f(x)_i = \frac{a_i}{x_i}$	Función de rentabilidad para cada intersección	Función de rentabilidad para cada flujo
a_i	Suma de las colas en la intersección i	Cola del flujo i en cada intersección
$\bar{f}(x) = \frac{1}{\sum x_i} \sum x_i f_i$	Rentabilidad promedio de la Red	Rentabilidad promedio de la intersección
$P = \sum x_i$	Tiempo Total de Verde para la Red	Tiempo total de verde para la intersección
$a = \sum a_i$	Sumatoria de todas las colas en la red	Sumatoria de las colas de los flujos de la intersección.

Ecuación 3. Sistema matricial de ecuaciones diferenciales para definir reparto de tiempo total de verde

$$E = \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \dots & 1 & 0 \\ 0 & 0 & \dots & 0 & 0 \end{bmatrix} \quad X' = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \vdots \\ \dot{x}_i \\ \vdots \\ \dot{x}_{i+1} \end{bmatrix} \quad A = \begin{bmatrix} -a/P & 0 & \dots & 0 \\ 0 & -a/P & \dots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \dots & -a/P & 0 \\ 1 & 1 & \dots & 1 & -1 \end{bmatrix} \quad X = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_i \\ \vdots \\ P \end{bmatrix} \quad B = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_i \\ \vdots \\ 0 \end{bmatrix}$$

Éste es un sistema de tantas ecuaciones como número de cruces haya en la red más la ecuación de restricción $\sum x_i = P$.

La solución de este sistema de ecuaciones, es decir los valores de x_i , para los valores iniciales de x_i y \dot{x}_i , los cuales son 0 y a_i respectivamente para la primera iteración, representan los tiempos de verde que se deben asignar a cada intersección.

En el nivel de jerarquía menor del algoritmo, se repite el proceso para encontrar el tiempo de verde para cada flujo del asignado a la intersección a la que pertenece. El sistema de ecuaciones es de la misma forma que para el caso anterior y viene representado matricialmente en la Ecuación 4.

Ecuación 4. Sistema matricial de ecuaciones diferenciales para definir reparto de tiempo de verde en una intersección

$$E = \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \dots & 1 \\ 0 & 0 & \dots & 0 \end{bmatrix} X' = \begin{bmatrix} \dot{x}_{i1} \\ \dot{x}_{i2} \\ \vdots \\ \dot{x}_{ij} \\ \dot{x}_{ij+1} \end{bmatrix} A = \begin{bmatrix} -a_i/x_i & 0 & \dots & 0 \\ 0 & -a_i/x_i & \dots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \dots & -a_i/x_i \\ 1 & 1 & \dots & 1 \end{bmatrix} X = \begin{bmatrix} x_{i1} \\ x_{i2} \\ \vdots \\ x_{ij} \\ x_i \end{bmatrix} B = \begin{bmatrix} a_{i1} \\ a_{i2} \\ \vdots \\ a_{ij} \\ 0 \end{bmatrix}$$

donde j representa el índice del flujo en la intersección i .

Éste es un sistema de tantas ecuaciones como número de flujos haya en cada intersección más la ecuación de restricción $\sum x_{ij} = x_i$

5.1.2. Tiempo total de verde para la red (P). El tiempo total de verde para la red (P) es el que se asigna a toda la red para que sea repartido entre las intersecciones de la misma. El tiempo **P** es definido a partir de un promedio de la ocupación de la red teniendo en cuenta datos tomados en el último ciclo o duración del último plan de señales ejecutado en la red. La ocupación de cada flujo se define de la siguiente manera:

Ecuación 5. Ocupación de un flujo o fila de una intersección

$$o_{ij} = \frac{a_{ij}}{\max A_{ij}}$$

Para lo cual **$\max A_{ij}$** es el valor máximo de la cola o valor de saturación del flujo **j** en el cruce **i**. Así, el valor de ocupación de la red es:

Ecuación 6. Ocupación promedio de la Malla

$$O = \frac{\sum_{i=1}^m \sum_{j=1}^n o_{ij}}{m * n}$$

Donde **m** y **n** son el número de cruces en la red y el número de flujos para cada cruce respectivamente.

Así, las condiciones tenidas en cuenta para determinar el tiempo **P** son las siguientes:

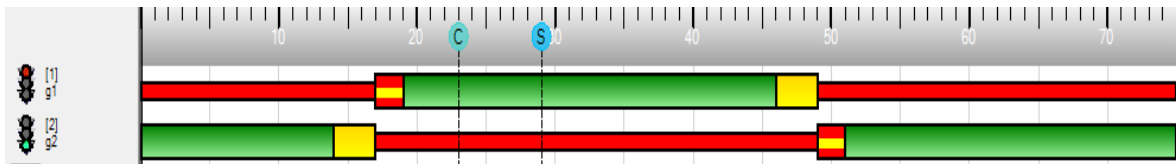
Cuadro 2. Condiciones para determinar tiempo total de verde para la Red

si O	entonces P =
$\geq 0 \text{ y } \leq 0.25$	45 seg x m
$> 0.25 \text{ y } \leq 0.50$	60 seg x m
$> 0.50 \text{ y } \leq 0.75$	75 seg x m
> 0.75	90 seg x m

5.1.3. Generación del plan de señales. El Plan de Señales se genera de acuerdo a la configuración o composición del cruce. Para este caso, todos los cruces tienen la configuración más sencilla: 2 grupos de señales vehiculares enemigos y ningún grupo de señales peatonal. Cuando se habla de amistad o enemistad de dos o más grupos de señales se refiere a la posibilidad que tienen éstos de estar en estado o luz verde al mismo tiempo.

Teniendo esto en cuenta, para cada grupo se toma el tiempo de verde entregado por el algoritmo y se añaden el tiempo de amarillo y rojo-amarillo estandarizados para este tipo de configuración en tres y dos segundos respectivamente.

Figura 10. Plan de Señales con dos grupos de señales vehiculares



Fuente: Software Cronos para diseño de planes de señales.

La figura 10 muestra un plan de señales con Duración de Ciclo de 75 segundos diseñado para una intersección compuesta por dos flujos enemigos. Además se observan dos señales: de conexión y de sincronismo. La señal de conexión es el instante en que comienza a ejecutarse un plan de señales predeterminado después de pasar por un plan de destello cuando un Equipo de Control ha sido reiniciado. Por otra parte, la señal de sincronismo es el instante en el que se genera el cambio de un plan a otro.

6. DESARROLLO DE LA APLICACIÓN MOGUT Y CONVERSIÓN DEL ALGORITMO DE CONTROL MULTINODAL AL LENGUAJE DE PROGRAMACIÓN C# (CSharp)

El optimizador de mallas de tráfico vehicular basado en inteligencia computacional recibe el nombre de MOGUT por sus siglas en inglés *Multinodal Optimization for Grids of Urban Traffic*. El MOGUT es una aplicación de software que busca optimizar el comportamiento del tráfico vehicular en una determinada red o malla urbana modificando continuamente el tiempo de verde para cada intersección y el reparto del mismo entre sus flujos con el fin de reducir las colas en modo global dentro de la red, logrando así que los usuarios experimenten un viaje más cómodo realizando menos paradas y/o esperando menor cantidad de tiempo en los semáforos.

El algoritmo de control desarrollado inicialmente por el Ingeniero Pablo Ñañez Ojeda de la Universidad de los Andes está escrito en el lenguaje de programación del software Matlab. Por esta razón, es necesario realizar la conversión del algoritmo al lenguaje de programación C# utilizando el programa Microsoft Visual Studio bajo la plataforma .NET por dos razones:

- Compatibilidad con la Central de Tráfico IMATIC – RTS puesto que ésta y sus comandos para comunicación con el equipo de control se encuentran desarrollados en dicho lenguaje.
- Todos los desarrollos de la empresa RTSotions están hechos sobre la plataforma .NET. Matlab es un software sujeto a su propietario The Mathworks y cuenta con su propio entorno de desarrollo, y aunque posee la herramienta "Application Deployment" para que sus funciones puedan ser utilizadas en ambientes de construcción .NET y Java, presenta una limitante y es que el computador o servidor donde se ejecute la aplicación necesita el MATLAB Component Runtime para que sus archivos funcionen bien.

La aplicación obedece al diagrama de implementación que se presenta en la figura 11. Los datos de las colas son enviados cada 10 segundos desde los controladores de campo *FC* (Field Controller) que representan a cada intersección hacia la Central. En la Central, como ya se explicó en la sección 5.1.1, se identifican dos jerarquías de control: una mayor en la que una instancia del controlador llamada RD_{total} (Replicator Dynamics) hace un reparto del tiempo de verde total entre cada uno de los cruces de la malla, recibiendo como entradas los índices de rentabilidad PI (Profitability Index) de cada cruce; y una jerarquía menor

en la que se instancia un controlador RD_i para cada controlador de campo que calcula el reparto de verde entre los flujos o filas de la intersección. Posteriormente, se genera un plan de señales que es comparado con los grabados en la base de datos SpDB (Signal Plan Data Base) que es una copia de la base de datos de la memoria del equipo de campo, y se escoge el plan más óptimo para ser ejecutado en cada controlador de acuerdo a un modo de selección. Este cálculo se hace cada 30 segundos, y se envía el comando de cambio de plan con el índice del plan a ejecutar para cada cruce a los correspondientes controladores de campo los cuales lo cumplen teniendo en cuenta condiciones de seguridad (marca de sincronismo).

Además, se describen más adelante en este capítulo, las etapas de Almacenamiento y Depuración de la base de datos de planes sugeridos. Esta es una base de datos que almacena planes que posiblemente podrían ser almacenados en la memoria del controlador de campo si así lo considera un operario.

6.1. CLASES DE LA APLICACIÓN EN C#

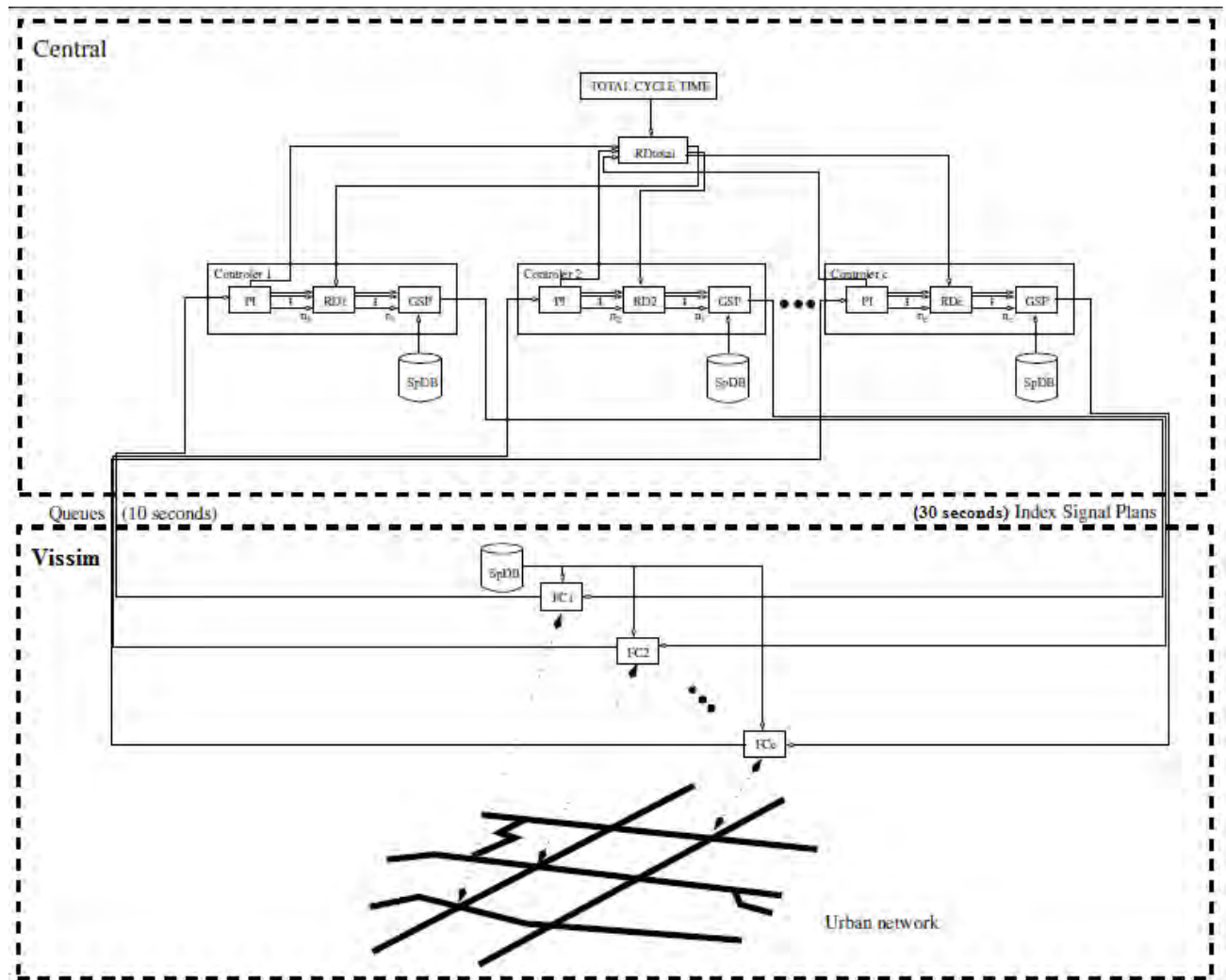
En el desarrollo de la aplicación en C# se pueden identificar las siguientes clases (Ver Anexo A):

6.1.1. Clase principal. Esta es la clase *main* que despliega la ventana principal de la interfaz gráfica (ver capítulo 9). En esta clase se ejecuta cada 30 segundos la rutina principal que actualiza los datos para el cálculo, instancia la clase *Calcular* y accede a ella para determinar los planes de señales a ejecutar, envía los cambios de planes a los equipos y depura la base de datos de planes sugeridos.

6.1.2. Clase calcular. Genera las matrices y parámetros necesarios para pasárselos al modelo de Replicator Dynamics con los datos recibidos de la *Clase Principal*. Además, tiene métodos para calcular los nuevos planes de señales y aplica la métrica de comparación con los planes almacenados en el equipo para definir un ganador y actualiza la base de datos de planes sugeridos con nuevos planes si así lo determina el algoritmo.

6.1.3. Clase controlador mallas. En esta clase se modela el controlador basado en Replicator Dynamics.

Figura 11. Diagrama de Implementación MOGUT



Fuente: ÑAÑEZ, Pablo., TRIANA, Carlos. Technical Report 3: Descripción Módulo MOGUT para la Central de Tráfico. Santiago de Cali, 2013. 1 archivo de computador.

6.1.4. Clase intergratorRK4V. En esta clase se aplica el método de Runge-Kutta de cuarto orden para la solución de ecuaciones diferenciales.

6.1.5. Clase configuración. Allí se definen todos los parámetros de configuración del algoritmo como modo de selección de planes, ponderación, extrapolación y el umbral de comparación para inclusión de un plan a la base de datos de planes sugeridos.

6.1.6. Clases mensaje y tipo. Estas clases se utilizan para definir cuál será la apariencia de los mensajes a desplegar en la ventana principal de monitoreo.

6.2. MEDICION DE LAS COLAS

La medición de las colas en la red se hace por medio de los sensores de detección de video PICO de ITERIS, INC., instalados en cada flujo de las intersecciones. En VISSIM, se dibujan directamente los sensores sobre la calle. En ambos casos se configuran tantos sensores virtuales a la salida y entrada de la cola como carriles tenga la vía. En este caso son dos de entrada a la cola y dos de salida de la cola (uno por cada carril). Los sensores de entrada, que son los ubicados lejos de la línea de parada, detectan la presencia de un vehículo al pasar sobre él y lo adiciona a la cola, mientras que los de salida que se encuentran justo antes de la línea de parada, los sustraen del valor de la cola.

En la figura 12.a se puede apreciar una imagen de una de las cámaras instaladas sobre la calle 82 en Barranquilla, donde se distinguen los cuatro sensores: P1 y P3 son los sensores de salida, y P2 y P4 son los de entrada. Estos mismos sensores se pueden visualizar en la figura 12b dibujados a la entrada y salida del flujo.

La transmisión de los valores de las colas se hace a través de comunicación GPRS a la Central de Tráfico cada diez segundos. En la Central, se almacenan las transmisiones en una base de datos desde la cual se extraen posteriormente para su utilización en el proceso de cálculos. La figura 13 muestra un ejemplo de la tabla en la que se almacenan dichos datos. La columna "**id_historicocola**" muestra un número de identificación único que se crea para cada transmisión; la columna "**codcruce**" despliega el código de cada cruce o intersección el cual es único para la ciudad; "**numeroflujo**" es el número que identifica el flujo de una intersección dentro del equipo. Para el caso que se observa en la figura, el cual fue una prueba de simulación realizada para tres cruces, un solo equipo controla

los tres cruces (2865, 2972, 2973), y cada cruce tiene dos flujos, por lo que se ven seis flujos transmitiendo. La columna **"cola"** muestra el valor de la cola para cada flujo de cada una de las intersecciones en el instante de la transmisión. Luego se encuentra la columna **"saturado"** la cual indica con un 1 si el flujo se encuentra saturado o un 0 si no lo está. La condición de saturación se determina cuando los sensores de entrada a la cola se encuentren en 1, es decir que haya presencia de vehículos sobre los sensores, y que el semáforo de ese flujo se encuentre en rojo. Por último, la columna **"datecreated"** indica la fecha y hora en la que se realizó la transmisión de los datos.

Figura 12. Sensores de detección a) ITERIS b) VISSIM



6.3. ALGORITMO DE CONTROL

El algoritmo es convertido al lenguaje C# para ser integrado a las demás etapas de la aplicación. Ver el capítulo 5 para la descripción del algoritmo de control basado en Replicator Dynamics.

Figura 13. Base de Datos, Tabla Histórico-Colas

id_historicocola	codcruce	numeroflujo	cola	saturado	datecreated
cae51b91-d9e7-11e1-861d-9cca1f3bcb4b	2865	1	0	0	2012-07-29 20:43:01
caee9216-d9e7-11e1-861d-9cca1f3bcb4b	2865	2	0	0	2012-07-29 20:43:01
caf817be-d9e7-11e1-861d-9cca1f3bcb4b	2972	3	0	0	2012-07-29 20:43:01
cb019cfc-d9e7-11e1-861d-9cca1f3bcb4b	2972	4	0	0	2012-07-29 20:43:01
cb0b260f-d9e7-11e1-861d-9cca1f3bcb4b	2973	5	0	0	2012-07-29 20:43:01
cb14a902-d9e7-11e1-861d-9cca1f3bcb4b	2973	6	0	0	2012-07-29 20:43:01
c4de9ac0-d9e7-11e1-861d-9cca1f3bcb4b	2865	1	0	0	2012-07-29 20:42:51
c4e81c26-d9e7-11e1-861d-9cca1f3bcb4b	2865	2	0	0	2012-07-29 20:42:51
c4f1a24f-d9e7-11e1-861d-9cca1f3bcb4b	2972	3	0	0	2012-07-29 20:42:51
c4fb2923-d9e7-11e1-861d-9cca1f3bcb4b	2972	4	0	0	2012-07-29 20:42:51
c504ac52-d9e7-11e1-861d-9cca1f3bcb4b	2973	5	0	0	2012-07-29 20:42:51
c50e338b-d9e7-11e1-861d-9cca1f3bcb4b	2973	6	0	0	2012-07-29 20:42:51
bf244f89-d9e7-11e1-861d-9cca1f3bcb4b	2865	1	0	0	2012-07-29 20:42:41
bf2dd57a-d9e7-11e1-861d-9cca1f3bcb4b	2865	2	0	0	2012-07-29 20:42:41
bf3757d2-d9e7-11e1-861d-9cca1f3bcb4b	2972	3	0	0	2012-07-29 20:42:41
bf40dcb3-d9e7-11e1-861d-9cca1f3bcb4b	2972	4	0	0	2012-07-29 20:42:41

6.4. COMPARACIÓN DE PLANES DE SEÑALES

Con el propósito de medir la similitud de dos soluciones (planes de señales) se plantea la distancia ε (epsilon). Esta métrica, desarrollada por el Ingeniero Pablo Ñañez Ojeda de la Universidad de los Andes en convenio con IMATIC INGENIERIA LTDA, tiene el objetivo de medir la similitud de dos planes con respecto a:

- Estructura
- Duración de las fases
- Secuencia
- Duración del ciclo

El plan de un grupo de señales puede ser analizado como un arco híbrido, es decir una solución cuyo dominio del tiempo está compuesto por dos componentes: el primero hace referencia al tiempo continuo en el cual el sistema evoluciona, y el segundo hace referencia a los saltos discretos del sistema. Es decir:

$$\varphi(t,j)$$

en donde, t hace referencia al tiempo continuo, j hace referencia al tiempo discreto, y φ es el valor de la señal en el tiempo (t,j) . Para poder cuantificar la

distancia de las señales es necesario asignar un número para cada estado del plan de señales, de la siguiente manera.

- $\varphi = 1$ cuando el estado es Rojo Amarillo.
- $\varphi = 2$ cuando el estado es Verde.
- $\varphi = 3$ cuando el estado es Verde Intermitente.
- $\varphi = 4$ cuando el estado es Amarillo.
- $\varphi = 5$ cuando el estado es Rojo.

La ε -cercanía, así como la distancia de Hamming, mide el cambio necesario para hacer una señal idéntica a la otra. Es decir, cual es el envolvente que alrededor de la señal inicial contiene la señal a medir. Entre menor sea la envolvente necesaria, más similares son las señales, entre mayor sea la envolvente, más difieren una de otra. Es importante aclarar que esta envolvente tiene que considerar tanto el tiempo continuo como el tiempo discreto.

Para todo $(t,j) \in \text{dom}(\varphi_1) \exists s, (s,j) \in \text{dom}(\varphi_2)$ tal que:

$$|t - s| = \varepsilon$$

ó

$$|\varphi_1(t,j) - \varphi_2(s,j)| = \varepsilon$$

El valor asignado a ε depende de cuál de las dos opciones sea la mayor ya que describe la mayor envolvente o distancia necesaria para que las dos señales sean idénticas.

Este proceso de comparación se aplica entre el plan generado tanto con los almacenados en la memoria del Equipo como con los planes almacenados en la base de datos de planes sugeridos. De los planes almacenados en el equipo, se considera el plan de señales ganador aquel que sea más similar al nuevo plan generado por el algoritmo, es decir, el que tenga el menor valor ε al ser comparado con éste. Este es el plan que se propone para ser ejecutado teniendo en cuenta los criterios expuestos en la sección 6.5.

6.5. SELECCIÓN DE PLANES A EJECUTAR

Con el propósito de satisfacer las diferentes necesidades que se puedan presentar en la malla o la flexibilidad que exista gracias a la geometría de las intersecciones individualmente y de la malla como un conjunto, se han definido distintas opciones al momento de que un plan sea escogido para ser ejecutado en el equipo de control.

6.5.1. Modo egoísta. En este modo la aplicación escoge el mejor plan de los almacenados en el equipo de control para cada cruce individualmente sin darle importancia a la decisión tomada para las demás intersecciones.

6.5.2. Modo compartir duración de ciclo. Para cada una de las intersecciones, se define un plan ganador por cada grupo de planes que presenten la misma duración de ciclo. Posteriormente se procede a agrupar el plan ganador por cada duración de ciclo en cada intersección con los ganadores de las demás intersecciones que tengan igual duración de ciclo a éste. Utilizando uno de los métodos de ponderación descritos en la sección 6.5.3 se definen los planes que serán ejecutados en el equipo asegurando de esta manera que todos los ciclos en la malla presenten la misma duración y no se genere un desfase inesperado.

6.5.2.1. Modo Compartir Duración de Ciclo y Reparto de Verde. Aunque el modo compartir duración de ciclo evita problemas como el desfase permanente en la malla, puede generar otros inconvenientes como el represamiento de vehículos en un corredor mientras otro quede completamente vacío y se desperdicie su tiempo de verde debido a las diferencias en los repartos en el plan de señales. Por este motivo, para asegurar que los cruces se encuentren completamente en fase y que se cumpla la premisa de la ola verde en el corredor principal, se ejecuta exactamente el mismo plan de señales en cada una de las intersecciones. Ese plan común a ejecutar se escoge de igual forma utilizando uno de los métodos de ponderación definidos (Ver sección 6.5.3). Para que este modo pueda funcionar se requiere que la base de datos de planes de señales de cada cruce sea una copia exacta de las bases de datos de los demás cruces de la malla. Esto quiere decir, que no solo deben ser los mismos planes de señales en composición y configuración, sino que también deben estar almacenados en la misma estructura y con el mismo número en cada intersección.

6.5.3. Modos de ponderación. Para seleccionar el plan a ejecutar para cada intersección, el plan ganador por cada duración de ciclo (*Modo Compartir Duración*

de Ciclo) o cada único plan almacenado en el controlador de campo (*Compartir Duración de Ciclo y Reparto de Verde*), se agrupa con sus equivalentes de las demás intersecciones de la red. Finalmente, se realiza una comparación entre estos diferentes grupos o configuraciones de planes de señales y, según el grupo de planes que sea considerado como el mejor, se envía al equipo la solicitud de cambio de planes. Esto se hace con el fin de lograr el mayor beneficio para la red en general.

Para agrupar los planes de cada intersección se calcula el valor b que está dado por la sumatoria de los índices ε obtenidos para cada uno de ellos en la etapa de comparación de planes de señales multiplicados por un valor de ponderación p . La agrupación se realiza como lo muestra la ecuación 7 donde el valor p está definido por uno de los dos modos de ponderación descritos en esta sección.

Ecuación 7. Cálculo del índice b para grupos de planes de señales ganadores para cada intersección

$$b = \sum_{j=0}^m \varepsilon_{i,j} * p$$

En esta ecuación i es el número del plan de señales en el equipo de control, y j es el índice del cruce en la malla.

Nuevamente, el conjunto de planes para cada intersección que tenga el menor valor para b será el escogido como ganador.

6.5.3.1. Un vehículo un voto. En este modo, cada vehículo de la malla representa un voto a favor de la intersección en la que se encuentra. El número de vehículos en el cruce se multiplica por el valor ε del plan para dicho cruce para indicar la prioridad o el peso que éste tiene al momento de tomar la decisión de los planes a ejecutar. De esta manera, el valor p de ponderación está dado por:

Ecuación 8. Cálculo del valor de ponderación p para el modo "un vehículo un voto"

$$p = \sum_{k=0}^n q_{k,j}$$

Donde n es el número de flujos o entradas que tiene la intersección j y q es el valor de la cola.

6.5.3.2. Un cruce un voto. Para este modo, cada cruce entrega un voto por el plan de señales que más le conviene a él. Por consiguiente, el valor de ponderación $p = 1$, puesto que cada plan ha sido elegido para un único cruce.

6.6. EXTRAPOLACIÓN DEL VALOR DE LA COLA EN ESTADO DE SATURACIÓN

Una vía, para este proceso de optimización, se considera saturada cuando se ha alcanzado el máximo de cola posible. En software, esto se mide detectando cuando el estado de la señal de control de encuentra en rojo y todos los sensores que están ubicados a la entrada de la cola están enviando un pulso constante al equipo, indicando que hay vehículos siendo detectados y estos no avanzan. Cuando esto ocurre, es posible que la cola siga creciendo tanto como el largo de la vía lo permita, pero por restricciones físicas en la ubicación de los sensores, es imposible medirla. Por esta razón, se desarrolla un método que permite estimar cual es el valor de la cola en el siguiente instante de muestreo al que se detectó una saturación. Este método es llamado *extrapolación*. Así, el valor de la cola en un instante de muestreo cuando la anterior muestra arrojó saturación, se define por la ecuación 9.

Ecuación 9. Cálculo de extrapolación del valor de la cola

$$q_h = q_{h-1} + (e * n_c * T)$$

Donde h es el número de la muestra, n_c es el número de carriles que tiene la calle, T es el periodo de muestreo (actualmente configurado en 10 segundos) y e es el valor de extrapolación que indica el estimado de vehículos que se suman a la cola

por carril por segundo. Este índice es configurable de acuerdo al análisis previo que haya hecho el usuario para la red.

6.7. ALMACENAMIENTO Y DEPURACIÓN DE PLANES DE SEÑALES SUGERIDOS

Los planes de señales generados por el algoritmo de control no son directamente ejecutados en el equipo de control, sino que son comparados con los ya almacenados en éste para ejecutar el más apropiado de acuerdo a los diferentes modos de selección (Ver sección 6.5). Por esta razón, es necesario desarrollar un método de almacenamiento de los planes sugeridos por el algoritmo de modo que puedan ser tenidos en cuenta para una posible inclusión de los mismos en la memoria del equipo de control. Así mismo, se debe diseñar un método de depuración de la base de datos de planes sugeridos para evitar una sobrecarga que colapsaría el programa.

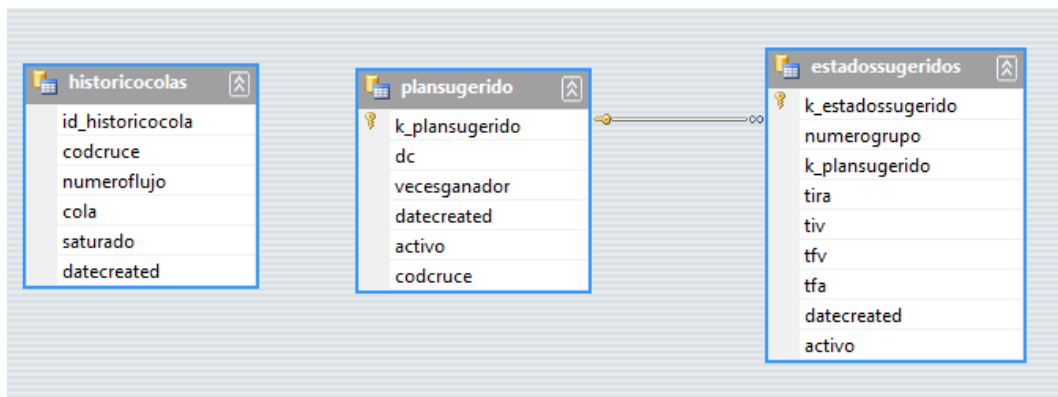
6.7.1. Almacenamiento de Planes de Señales Sugeridos. Puesto que almacenar todos los planes de señales sugeridos por el algoritmo en cada iteración (cada 30 segundos) significaría un gasto de tiempo de ejecución y de la memoria de la central, para decidir si un plan se almacena se revisa si el valor o distancia ϵ resultante de la comparación del plan sugerido con el existente a ejecutar es mayor o menor que un valor predeterminado. Si la distancia es menor a éste, el plan existente se considera lo suficientemente parecido al sugerido como para generar el impacto que se desea en la malla vial y por lo tanto no se almacena; pero, si la distancia es mayor a este valor se decide almacenarlo para revisión y posible inclusión en el equipo, puesto que se considera que ninguno de los planes existentes para dicho cruce creará el comportamiento tal y como se desea en la malla.

Adicional al criterio anteriormente mencionado, los planes de señales sugeridos por el algoritmo se comparan también con los almacenados en la base de datos de planes sugeridos para revisión, de su mismo cruce, y utilizando el mismo valor predeterminado, si el nuevo plan sugerido es muy similar a alguno de los allí almacenados (utiliza métrica de comparación), se incrementa un contador llamado **"vecesganador"** para dicho plan almacenado y no se almacena el nuevo. Así, en una posterior revisión por una persona encargada se analiza la fecha de creación de cada plan de señales sugerido en la base de datos y el valor del contador **"vecesganador"** con el fin de determinar si debe ser incluido en el equipo, puesto que un valor alto de dicho contador significaría que un mismo comportamiento se está repitiendo en la malla vial que requiere ese plan.

Esta estrategia de almacenamiento refleja una retroalimentación del sistema y permite que la selección de planes con el tiempo sea más óptima y flexible.

6.7.2. Depuración de la base de datos de planes de señales sugeridos. Es necesario depurar continuamente la base de datos para evitar que se sature y ocasione que el programa se vuelva muy lento y no le permita ver al usuario los datos de ejecución en tiempo real o, que en el peor de los casos, colapse.

Figura 14. Diagrama Entidad - Relación



Puesto que la memoria del equipo de control soporta hasta 128 planes de señales almacenados, se busca mantener este mismo número de planes en esta base de datos para cada cruce o intersección en la malla vial. El proceso de actualización de la base de datos se hace cada 30 segundos cuando se ejecuta el algoritmo de control, pero el proceso de depuración se hace solo cada 2 horas con el fin de darle tiempo al algoritmo de tener una buena cantidad de planes almacenados. Si hay más de 32 planes almacenados por cada cruce se procede a la depuración para la cual se ordenan los datos de la base de datos de acuerdo al siguiente criterio:

Ecuación 10. Criterio para depuración de la base de datos de planes sugeridos

$$c = \frac{v_g}{1 + Diff_Hr}$$

Donde **c** es el valor del criterio de ordenamiento, **v_g** es el valor de la variable "vecesganador", y **Diff_Hr** es la diferencia entre la Fecha-hora de inclusión del plan a la base de datos y la Fecha-hora actual. Esta diferencia entrega un valor entero que indica el número de horas que han transcurrido desde que se creó el plan en la base de datos. Se adiciona un 1 a esta diferencia para eliminar el error que puede surgir cuando la diferencia sea 0.

Así, este criterio indica el número promedio de veces que un dicho plan ha resultado ganador por hora al ser comparado con un nuevo plan de señales sugerido frente a los demás que se encuentran en la base de datos. De esta manera, se eliminan los planes con menor valor de *criterio* de modo que se mantengan siempre 32 planes por cada cruce.

7. VALIDACIÓN DEL ALGORITMO DE CONTROL POR MEDIO DE SIMULACIÓN HARDWARE-IN-THE-LOOP (HIL)

Se realizaron pruebas de simulación con la ayuda del software VISSIM para micro-simulación de tráfico de la PTV Vision con el fin de medir el desempeño del algoritmo de control. Se utiliza este software por las facilidades que posee de interconexión con el equipo de control, sus variadas herramientas de configuración de simulación como flujo vehicular, tipo de vehículos, tamaño de la red, velocidad, simulación de los sensores inductivos, etc., además de la facilidad en la medición de los diferentes índices de desempeño, lo que permite hacer un acercamiento bastante acertado a la realidad.

7.1. MONTAJE PARA LAS PRUEBAS

7.1.1. Descripción de las partes. A continuación se presentan cada una de las partes que conforman el montaje para la realización de las pruebas de simulación.

7.1.1.1. Equipo de control. Equipo desarrollado por IMATIC INGENIERIA Ltda., que cuenta con un sistema operativo con microprocesador. Es el encargado de administrar las funciones de los diferentes grupos o semáforos en las intersecciones. Algunas de sus características son:

- Permite configurar desde 2 hasta 32 grupos vehiculares o peatonales.
- Administra 4 intersecciones totalmente de manera independientes.
- La capacidad de comunicación ha sido incrementada con la adición de multiprocesamientos, haciendo posible el desarrollo de tareas en tiempo real.
- Permite configurar hasta 32 entradas para sensores.

Estas y otras características hacen que el equipo sea muy flexible y le permiten adaptarse a diferentes configuraciones.

7.1.1.2. MOGUT. Aplicación desarrollada y descrita en el capítulo 6. Ésta se encuentra instalada en el mismo servidor donde está la central de tráfico. El servidor y el equipo de control se comunican por GPRS.

7.1.1.3. Central de tráfico IMATIC¹⁶. La aplicación Central de Tráfico IMATIC – RTS es una herramienta de administración y gestión del tráfico que permite la programación y la supervisión de los equipos de semaforización.

La configuración de los equipos está compuesta de una jerarquía que debe definirse en un orden estricto. Dicha jerarquía se define a continuación:

- Equipo: Define las características generales del equipo de semaforización, con información referente a la identificación del equipo al interior de la red de datos.
- Cruce: Define un conjunto de grupos que están asociados a una intersección.
- Grupo: Constituye un semáforo (entendido como el conjunto de luces).
- Estructura: Define el conjunto de características de los flujos vehiculares en el cruce (secuencia de verdes de los grupos).
- Plan: Define los intervalos de tiempo de la secuencia de cambios de los grupos de un cruce.
- Horario del plan: Indica el horario en el que se encuentra activo un plan.
- Horarios del ventilador: Indica el horario de encendido del ventilador del equipo.

Permite además configurar cada flujo de la red asignándolos a sus respectivos grupos de señales e indicando cuales sensores pertenecerán a cada uno de ellos.

Desde la Central se pueden enviar diferentes tipos de comandos a los equipos de control que se encuentran en la red entre los que se encuentra el comando de Programación del Equipo por medio del cual se da una orden de reprogramación con toda la información de configuración anteriormente descrita. La comunicación es completamente inalámbrica y utiliza la red celular así que no es necesario estar cerca del equipo para este fin.

7.1.1.4. VISSIM 5.40. Es una herramienta importante de simulación microscópica para el modelado de flujos de tráfico multimodal. Proporciona las condiciones

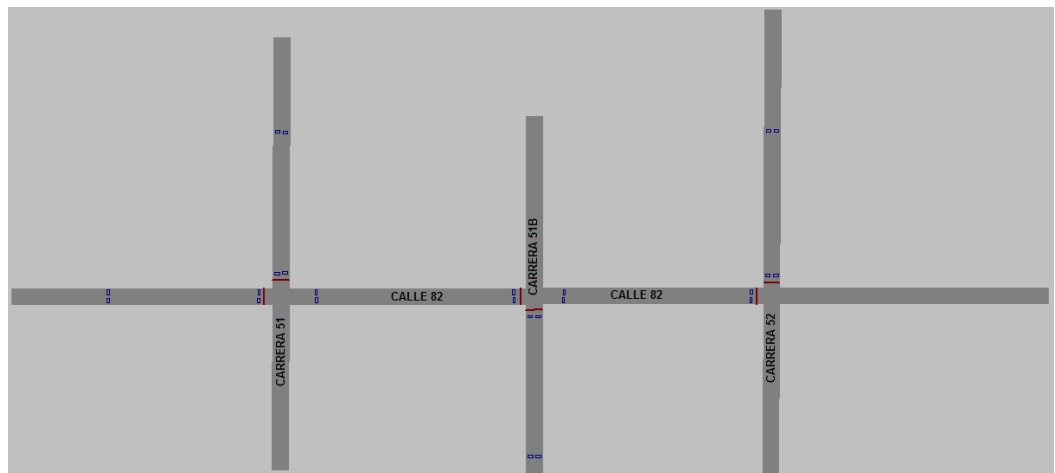
¹⁶ Soluciones en Tiempo Real Limitada. Central de Tráfico IMATIC – RTS: Manual de Usuario. [consultado 24 de Noviembre de 2011]. Disponible en internet: http://www.rts.com.co/Central/Manual_Central_V2.pdf

ideales para probar diferentes escenarios de tráfico de una manera realista y muy detallada antes de la implementación final.

Para un primer tipo de prueba, se simuló en VISSIM una red de 3 intersecciones (ver figura 15). Esta red permite probar el desempeño de la red bajo el efecto del control del MOGUT bajo condiciones ideales de tráfico donde el flujo vehicular sobre un corredor depende única y exclusivamente del flujo de entrada configurado para dicho corredor y no de los vehículos provenientes de otros cruces. Las intersecciones seleccionadas hacen parte de la malla sobre la cual se implementará el nuevo sistema de control en la ciudad de Barranquilla siendo estas intersecciones la calle 82 con carreras 51, 51B y 52.

Para el segundo tipo de prueba, se simuló en VISSIM una red de 6 intersecciones (ver figura 16) con la misma geometría de la red real de Barranquilla sobre la cual será implementado el MOGUT. Esta red corresponde a las calles 82 y 84 con carreras 51, 51B y 52.

Figura 15. Malla para simulación HIL 3 intersecciones – Calle 82 Carreras 51, 51B y 52



7.1.1.5. Hardware de acople. Sistema electrónico diseñado para convertir los niveles de voltaje de las señales entre el Equipo de Control y la tarjeta de adquisición de datos NI 6509, puesto que el Equipo de Control tiene salidas a un nivel de voltaje de 120 VAC y entradas a 24 VDC, mientras que las entradas y salidas de la DAQ trabajan a 5V DC.

Figura 16. Malla para simulación HIL 6 intersecciones – Calle 82 y Calle 84 con Carreras 51, 51B y 52

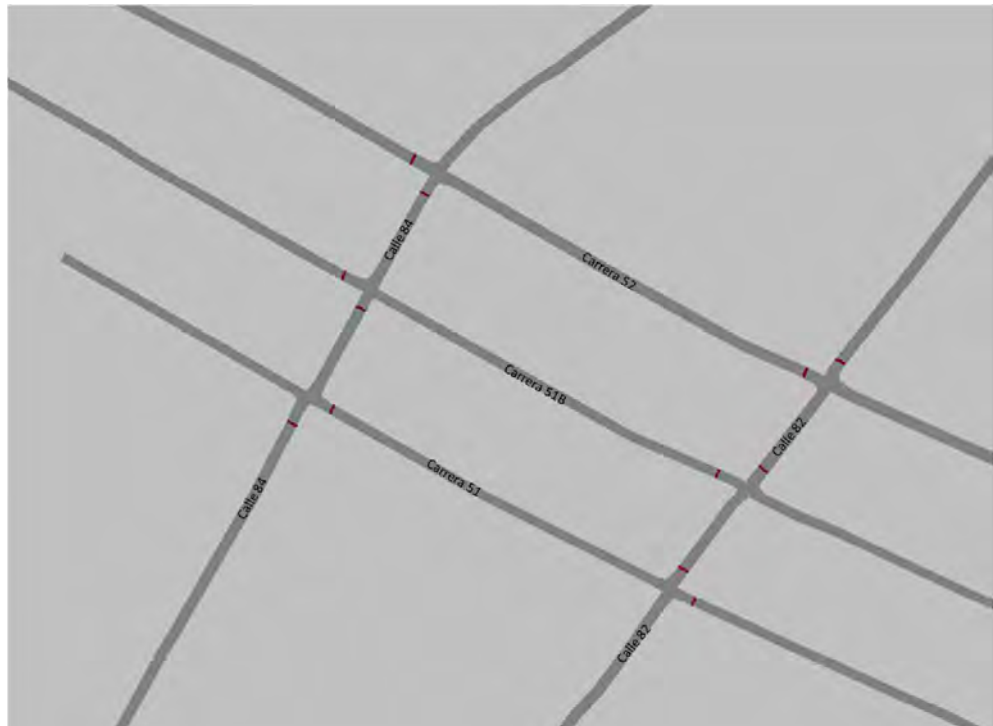


Figura 17. Acople de señales 120 VAC a 5 VDC

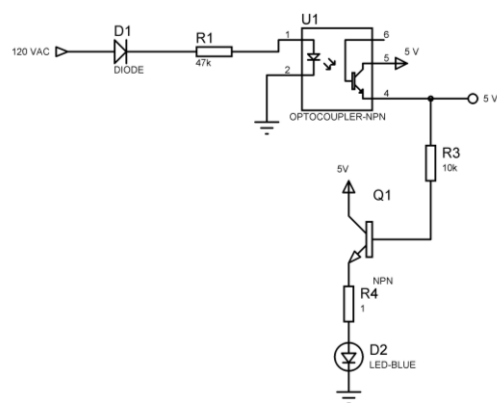
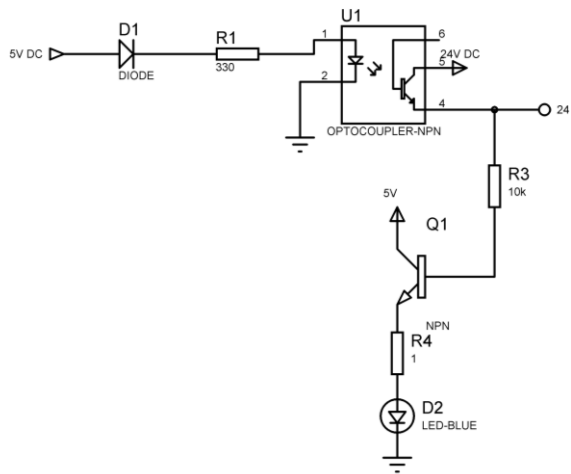


Figura 18. Acople de señales 120 VAC a 5 VDC

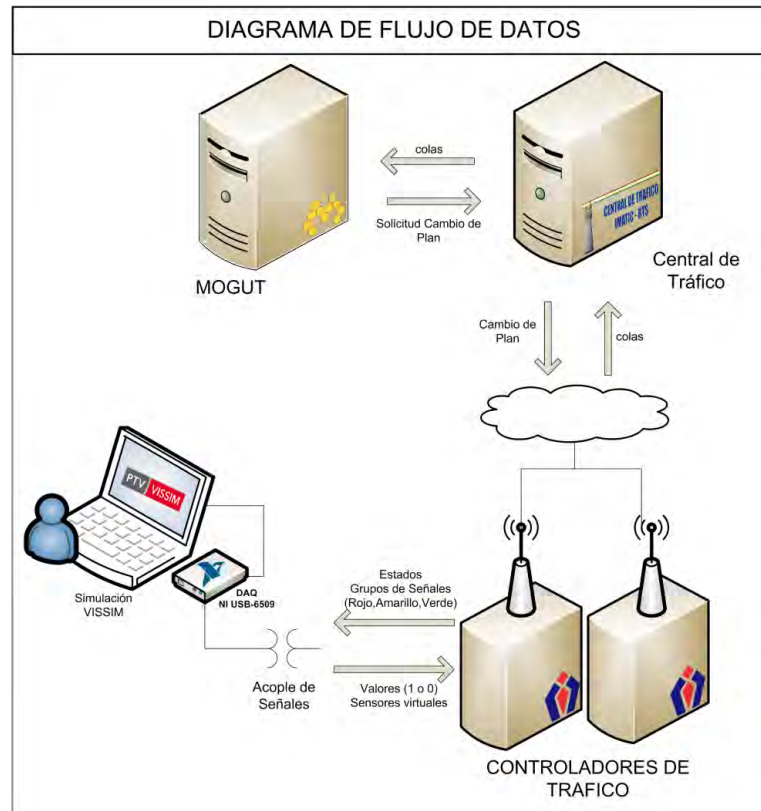


7.1.1.6. Tarjeta de adquisición de datos (NI USB-6509). El NI USB-6509 es un dispositivo de adquisición de datos industrial y portable con 96 canales de E/S digitales. Es compatible con TTL, CMOS, y niveles lógicos digitales de 5V. Cada puerto de 8 líneas es configurable individualmente como entradas o salidas. Puedes manejar dispositivos digitales externos, como relevadores de estado sólido (SSRs), con corriente de hasta 24 mA por línea usando la potencia disponible por USB. En el modo de alimentación por bus de datos, la USB-6509 puede proporcionar hasta 215mA en total para todas las líneas de salida digitales. Utiliza la fuente opcional de 12 VDC para proporcionar corriente adicional. Una USB-6509 es ideal para aplicaciones de adquisición de datos de propósito general así como para control industrial y pruebas de manufactura automatizadas. Ver Datasheet en Anexo B.

7.1.2. Descripción del montaje. El software de simulación VISSIM envía los pulsos generados por los sensores inductivos simulados en la red que indican presencia de vehículo. El equipo recibe los datos del software y calcula el valor de la cola el cual es enviado cada diez segundos a la central de tráfico donde es almacenado en una base de datos. A su vez el MOGUT, haciendo uso de la información almacenada en la central, calcula cada 30 segundos el plan más óptimo que se debe ejecutar para cada intersección de la red y lo envía al Equipo utilizando los comandos del servicio web instalado también en la Central de Tráfico. Las salidas del equipo de control que van normalmente conectadas a los semáforos se conectan al computador a través de un sistema electrónico que

convierte los niveles de voltaje y la Tarjeta de Adquisición de Datos NI USB-6509. Estos datos son interpretados por el VISSIM y ejecutados en la red simulada.

Figura 19. Diagrama de Interconexión



7.2. DISEÑO DE LAS PRUEBAS

Con el propósito de medir el desempeño de la aplicación software de optimización MOGUT se diseñan dos tipos de pruebas: una con una con un corredor de 3 intersecciones y otra con una malla de 6 intersecciones. El primer tipo de prueba busca medir el efecto de la optimización en una configuración simple de un solo corredor principal que es atravesado por tres calles y donde los vehículos viajan solamente en línea recta, simulando condiciones ideales del tráfico urbano. El segundo tipo de prueba pretende identificar qué efecto tiene el MOGUT directamente sobre una malla con características geométricas y configuraciones

iguales a las de la malla real sobre la cual será implementado el MOGUT en Barranquilla.

7.2.1. Prueba de simulación HIL para configuración de tres intersecciones.

Para esta configuración (Ver Figura 15) se diseñaron 3 experimentos que se ejecutan tanto en lazo abierto (sistema de control con plan de Señales fijo) como en lazo cerrado (Sistema de control MOGUT). A continuación se describen las pruebas a realizar.

Cuadro 3. Características prueba de Simulación HIL para configuración de tres intersecciones

Experimentos	Intervalos de Tiempo (mins)	Flujos de Entrada (Veh/h)			
		Corredor 1 (calle 82)	Corredor 2 (Cra 51)	Corredor 3 (Cra 51B)	Corredor 4 (Cra 52)
Experimento 1	0 - 60	2200	900	900	900
Experimento 2	0 - 30	2200	900	1500	900
	30 - 60	2200	1500	900	1500
Experimento 3	0 - 14	2200	1500	900	1500
	14 - 16	2200	1500	1500	1500
	16 - 30	1200	1500	1500	1500
	30 - 44	1200	900	900	900
	44 - 46	1200	900	1500	900
	46 - 60	2200	900	1500	900

Experimento 1. Para esta prueba el Corredor 1, que representa la calle 82 de la ciudad de Barranquilla, se encuentra saturado con un flujo de entrada de vehículos constante. Las tres vías secundarias (carreras 51, 51B y 52) presentan flujo constante también pero muy por debajo del límite de saturación que, de acuerdo a datos entregados por Barranquilla, se encuentra en aproximadamente 1300 veh/h.

Experimento 2. El corredor principal se encuentra saturado a un flujo de entrada de vehículos constante, mientras los corredores 1 y 3 presentan un flujo constante no saturado la primera mitad de la prueba y uno de saturación la segunda mitad.

Experimento 3. Se presentan dinámicas diferentes en cada uno de los corredores en cuanto al flujo de entrada de los vehículos. Esto se hace con el fin de representar variaciones inesperadas que pueden ocurrir en la dinámica del tráfico y que son casi imposibles de modelar puesto que no hacen parte del comportamiento común del tráfico en esa zona. Este tipo de comportamientos se pueden presentar por diferentes motivos: cierres de vías alternas, accidentes, daños en la carretera, eventos especiales en la ciudad, etc.

- **Características generales de los experimentos en lazo abierto**
- **Duración:** 60 minutos.
- **Duración de ciclo del plan de señales:** 60 segundos
- **Reparto de Verde:** 50% del tiempo de verde del ciclo para cada vía.
- **Características Generales de los experimentos en lazo cerrado**
- **Duración:** 60 minutos.
- **Modo Ponderación de Planes:** Un vehículo un voto.
- **Modo de Selección de Planes:** Comparten DC y Reparto
- **Extrapolación:** 0.5 veh/c*s
- **Umbral de comparación:** 15.

7.2.2. Prueba de simulación HIL para configuración de 6 intersecciones. En esta configuración (Ver Figura 16) se diseñó un solo experimento suficiente para modelar la dinámica que presenta comúnmente una malla de este tipo en un área urbana. El Cuadro 4 describe las características del experimento.

Cuadro 4. Características prueba de Simulación HIL para configuración de tres intersecciones

Experimentos	Intervalo de Tiempo (mins)	Flujos de Entrada (Veh/h)				
		Corredor 1 (Calle 82)	Corredor 2 (Calle 84)	Corredor 3 (Cra 51)	Corredor 4 (Cra 51B)	Corredor 5 (Cra 52)
Experimento 1	0 - 60	2200	2200	1300	1300	1300

Para esta prueba se configuran flujos de entrada de vehículos altos de modo que puedan saturar la red, simulando de esta manera el estado más crítico en el que se puede encontrar la misma.

- **Características generales del experimento en lazo abierto**
- **Duración:** 45 minutos.
- **Duración de ciclo del plan de señales:** 60 segundos
- **Reparto de Verde:** 50% del tiempo de verde del ciclo para cada vía.
- **Características generales del experimento en lazo cerrado**
- **Duración:** 45 minutos.
- **Modo Ponderación de Planes:** Un vehículo un voto.
- **Modo de Selección de Planes:** Comparten DC y Reparto
- **Extrapolación:** 0.5 veh/c*s
- **Umbral de comparación:** 15.

8. RESULTADOS Y ANÁLISIS

En este capítulo se presentan los resultados obtenidos a partir de las pruebas realizadas con cada sistema de control y la evaluación de los índices de desempeño. Esto se convierte en el argumento que permite decidir cuál de los dos sistemas de control evaluados es el más efectivo.

8.1. INDICES DE DESEMPEÑO

Un índice de desempeño es una variable cuyo valor permite hacer una comparación de la respuesta de dos o más controladores al enfrentarse a una misma prueba. Al mismo tiempo, esta comparación permite identificar la efectividad de un sistema de control sobre otro. Es necesario tener en cuenta varios índices de desempeño dependiendo del tipo de planta o sistema que se está controlando con el fin de minimizar el error al momento de decidir cuál controlador usar finalmente a partir de los resultados arrojados por la prueba. En este caso en particular, el software de simulación arroja los resultados del desempeño de la red y el valor de cada uno de los índices de desempeño, lo que hace estas mediciones mucho más confiables. Los índices escogidos a medir para esta prueba son los siguientes:

8.1.1. Tiempo de retraso promedio por vehículo (TRPV). Este índice indica en promedio cuantos segundos tuvo de retraso cada vehículo al cruzar la malla a partir del tiempo que VISSIM calcula debería demorarse un vehículo en hacer dicho recorrido bajo condiciones ideales (sin congestiones, sin paradas por semáforos, etc.)

8.1.2. Número promedio de paradas por vehículo (NPPV). Informa en promedio cuantas veces se detuvo cada vehículo durante su paso por la malla.

8.1.3. Distancia total viajada (DTV). Es la suma, en kilómetros, de las distancias viajadas por cada uno de los vehículos que cruzó la red durante el tiempo de simulación.

8.1.4. Tiempo total de viaje (TTV). Indica la suma en horas del tiempo que le tomó a cada vehículo cruzar la red durante el tiempo de simulación.

8.1.5. Tiempo total de retraso (TTR). Es la suma de los retrasos experimentados por cada uno de los vehículos que cruzó la red durante el periodo de simulación.

8.1.6. Velocidad promedio (VP). Es la relación entre los índices DTV y TTV que indica la velocidad promedio a la que viajó cada vehículo.

8.2. RESULTADOS

8.2.1. Resultados esperados. De los resultados de las pruebas HIL se espera que los índices de desempeño TRPV y NPPV presenten un decremento para la prueba con el MOGUT con respecto a la prueba en lazo abierto puesto que esto indicaría que los vehículos cruzaron más rápido la red y además el hecho de hacer menos cantidad de paradas, indica que la experiencia en la malla fue mejor.

Se espera también que el índice VP incremente, es decir que, que el flujo de los vehículo en la malla fue más veloz.

Los índices DTV y TTV podrían variar, pero en este caso es la combinación de ambos (VP) la que tiene mayor interés. Se presentan como resultados ya que justifican el valor de la Velocidad Promedio y pueden dar una idea del comportamiento experimentado en la red.

8.2.2. Resultados obtenidos. Después de realizadas las pruebas tanto con la red de 3 cruces como con la red de 6, se obtuvieron los siguientes resultados para cada uno de los índices de desempeño descritos en la sección 8.1.

8.2.2.1. Análisis de resultados para pruebas con red de 3 intersecciones. En el cuadro 5 se observa la comparación de los resultados arrojados para cada prueba de simulación HIL para los dos sistemas de control implementados.

En las tres pruebas se identifican los efectos esperados descritos en la sección 8.2.1. Los índices de desempeño TRPV, NPPV y TTR presentan un decremento de la prueba en lazo abierto a la prueba en lazo cerrado, lo que indica que el MOGUT tiene una mayor efectividad al momento de controlar el tráfico en las diferentes condiciones a las que fue sometida la malla. Aunque los dos primeros índices no presentan una variación muy grande cuando se ven en sus unidades originales y no en porcentajes, debe considerarse que estos valores son un

promedio por vehículo, pero al sumar esta ganancia para cada vehículo se puede notar que el desempeño que tiene la malla en general es bastante bueno.

Cuadro 5. Resultados de pruebas de simulación HIL con 3 intersecciones

	Sistema de Control	Indices de Desempeño					
		TRPV (s)	TTR (h)	NPPV	DTV (km)	TTV (h)	VP (km/h)
Prueba 1	<i>Lazo Abierto (Plan de Señales Fijo)</i>	31.972	40.374	1.105	1343.377	65.993	20.356
	<i>Lazo Cerrado (MOGUT)</i>	26.770	35.887	0.787	1479.767	64.199	23.050
	<i>Diferencia</i>	-5.202	-4.487	-0.318	136.390	-1.794	2.694
		-16.3%	-11.1%	-28.8%	10.2%	-2.7%	13.2%
Prueba 2	<i>Lazo Abierto (Plan de Señales Fijo)</i>	29.128	45.122	0.999	1530.421	73.891	20.712
	<i>Lazo Cerrado (MOGUT)</i>	28.75	44.672	0.754	1590.51	75.568	21.047
	<i>Diferencia</i>	-0.378	-0.45	-0.245	60.089	1.677	0.335
		-1.3%	-1.0%	-24.5%	3.9%	2.3%	1.6%
Prueba 3	<i>Lazo Abierto (Plan de Señales Fijo)</i>	25.068	36.091	0.79	1387.558	62.617	22.159
	<i>Lazo Cerrado (MOGUT)</i>	23.156	33.087	0.693	1361.959	59.09	23.049
	<i>Diferencia</i>	-1.912	-3.004	-0.097	-25.599	-3.527	0.89
		-7.6%	-8.3%	-12.3%	-1.8%	-5.6%	4.0%

Por otro lado, el índice que indica la velocidad promedio de cada vehículo también cumplió con las expectativas planteadas en la sección 8.2.1. En los tres experimentos se presenta un desempeño mejor con el control del MOGUT. Aunque en algunos casos, como la prueba No. 3, los índices DTV y TTV disminuyen lo que haría pensar que hubo un menor desempeño como resultado a la acción de control del MOGUT, al observar bien, se puede identificar que el decremento en el DTV fue muy mínimo comparado con el decremento del índice TTV, y por esta razón al relacionarse resulta una VP mayor. Por esta razón, es importante identificar el significado de cada índice de desempeño y determinar cuáles tienen mayor relevancia de acuerdo a lo que se desee medir.

8.2.2.2. Análisis de resultados para pruebas con red de 6 intersecciones. El Cuadro 6 registra los resultados obtenidos para el experimento con la red de 6 intersecciones implementando cada controlador (MOGUT y control con tiempos fijos).

Cuadro 6. Resultados pruebas de simulación HIL con 6 intersecciones

	Sistema de Control	Indices de Desempeño					
		TRPV (s)	TTR (h)	NPPV	DTV (km)	TTV (h)	VP (km/h)
Experimento 1	<i>Lazo Abierto (Control de Tiempos Fijos)</i>	93.331	162.862	1.904	8822.928	303.977	29.025
	<i>Lazo Cerrado (MOGUT)</i>	77.048	147.354	1.661	9767.378	303.744	32.157
	<i>Diferencia</i>	-16.283 -17.4%	-15.508 -9.5%	-0.243 -12.8%	944.45 10.7%	-0.233 -0.1%	3.132 10.8%

Se obtuvieron los resultados esperados para todos los índices de desempeño medidos. En comparación con los resultados para las pruebas con una red de tres intersecciones (ver Cuadro 5), éstos presentan mejoras mayores del control realizado por el MOGUT con respecto al realizado con tiempos fijos, lo que permite concluir que el MOGUT reacciona bastante bien tanto a la dinámica que se presenta en un corredor que atraviesa varias intersecciones como ante el comportamiento de una malla de tráfico urbano más compleja donde las decisiones de movimiento de los vehículos son menos predecibles.

9. INTERFAZ DE INTEGRACIÓN DEL MOGUT CON LA CENTRAL DE TRÁFICO

La interfaz es una herramienta que integra el sistema de control MOGUT con la información de la central de tráfico. Le permite al usuario configurar el funcionamiento del MOGUT seleccionando los diferentes modos de operación y parámetros necesarios. Además tiene una ventana de visualización que le permite al usuario monitorear los diferentes evento del programa como los datos de las colas en cada flujo de la malla, los cambios de planes a ejecutar, los planes de señales que se encuentra ejecutando el equipo en ese momento, y adicional a eso, una ventana donde visualiza los planes que continuamente sugiere el MOGUT para que sean analizados por una persona encargada y posiblemente almacenados en la memoria del equipo de control de tráfico.

Figura 20. Interfaz de usuario MOGUT – CENTRAL DE TRÁFICO

FrmPrincipal

Configuración

Monitoreo Colas y Planes Actuales

Monitoreo Planes Sugeridos

Registro de eventos

Plan a Enviar2013-02-07 13:28:21
cruce = 8003 Estructura = 1 Plan = 1 indice = 4

Plan a Enviar2013-02-07 13:28:21
cruce = 8002 Estructura = 1 Plan = 4 indice = 5

Plan a Enviar2013-02-07 13:28:21
cruce = 8001 Estructura = 2 Plan = 2 indice = 7

Tiempo de Verde para la Red2013-02-07 13:28:21
150

Ocupacion Flujo2013-02-07 13:28:21
cruce = 8003 flujo = 6 indice Desempeno = 0,11 saturado : NO

Ocupacion Flujo2013-02-07 13:28:21
cruce = 8003 flujo = 5 indice Desempeno = 0,39 saturado : NO

Ocupacion Flujo2013-02-07 13:28:21
cruce = 8002 flujo = 4 indice Desempeno = 0,22 saturado : NO

Ocupacion Flujo2013-02-07 13:28:21
cruce = 8002 flujo = 3 indice Desempeno = 0,5 saturado : NO

Ocupacion Flujo2013-02-07 13:28:21
cruce = 8001 flujo = 2 indice Desempeno = 0,33 saturado : NO

Ocupacion Flujo2013-02-07 13:28:21
cruce = 8001 flujo = 1 indice Desempeno = 0,83 saturado : NO

Plan a Enviar2013-02-07 13:27:51
cruce = 8003 Estructura = 1 Plan = 3 indice = 7

Plan a Enviar2013-02-07 13:27:51
cruce = 8002 Estructura = 1 Plan = 8 indice = 4

Colas

Drag a column header here to group by that column.

id	codcruce	numeroflujo	cola	saturado	datecreated
8001	2	5	0	2013-02-07 13:28:15	
8002	3	7	0	2013-02-07 13:28:15	
8002	4	3	0	2013-02-07 13:28:15	
8003	5	7	0	2013-02-07 13:28:15	
8003	6	1	0	2013-02-07 13:28:15	
8001	1	3	0	2013-02-07 13:28:06	
8001	2	6	0	2013-02-07 13:28:06	
8002	3	9	0	2013-02-07 13:28:06	
8002	4	4	0	2013-02-07 13:28:06	
8003	5	3	0	2013-02-07 13:28:06	
8003	6	1	0	2013-02-07 13:28:06	
8001	1	8	0	2013-02-07 13:27:55	
8001	2	4	0	2013-02-07 13:27:55	
8002	3	8	0	2013-02-07 13:27:55	
8002	4	1	0	2013-02-07 13:27:55	
8003	5	0	0	2013-02-07 13:27:55	
8003	6	0	0	2013-02-07 13:27:55	
8001	1	15	0	2013-02-07 13:27:47	

Planes Actuales

Drag a column header here to group by that column.

id	codigo	numeroestructura	numeroplan	duracion(ciclo)
8001	1	1	6	30
8002	1	6	6	60
8003	1	6	6	60

La figura 20 muestra la interfaz en funcionamiento donde se pueden identificar las diferentes partes: la ventana de visualización ubicada al costado izquierdo de la imagen es la ventana de monitoreo de eventos como cambios de plan, estado de cada flujo, etc.; la ventana superior derecha muestra los datos de las colas muestreados cada diez segundos; y, finalmente, la ventana inferior izquierda despliega los planes que actualmente ejecuta el equipo de control. La marca "máximo" permite visualizar la cola máxima transmitida durante el ciclo del último plan de señales ejecutado en lugar del valor de la cola instantáneo. La pestaña *configuración* despliega diferentes opciones para la parametrización del programa. En el Anexo C se puede apreciar más a detalle la descripción de cada una de las funciones de la interfaz gráfica.

10. CONCLUSIONES

- Se hizo un análisis detallado de las partes que comprendía el algoritmo de control multinodal realizado por el personal de la Universidad de los Andes en convenio con la empresa IMATIC INGENIERIA LTDA. basado en un modelo de inteligencia computacional llamado *Replicator Dynamics*. Se hizo una descripción del análisis matemático y conceptual de dicho modelo y como éste se puede aplicar al desarrollo de aplicaciones de control de tráfico. Se expone también la métrica de comparación utilizada para determinar la similitud entre dos planes de señales, la cual hace parte del trabajo desarrollado por la Universidad de los Andes.
- Se realizó el proceso de conversión del algoritmo basado en *Replicator Dynamics* del lenguaje de programación propio de Matlab al lenguaje CSharp que hace parte del paquete de programación de Microsoft Visual Studio. Se hizo una conversión conceptual y funcional del algoritmo puesto que hacer una conversión literal comprendía el uso de funciones de Matlab, lo cual no era factible por las condiciones que requería. Al mismo tiempo de la conversión del algoritmo y la métrica de comparación se desarrollaron otras etapas de procesamiento que juntos a estas dos conformaban la aplicación de software MOGUT (Multinodal Optimization for Grids of Urban Traffic).
- Se realizaron pruebas de simulación Hardware-in-the-loop con el fin de validar el correcto funcionamiento y desempeño del MOGUT y por consiguiente del algoritmo de control. Fueron diseñados dos tipos de pruebas: una para un corredor con tres intersecciones y otra para una malla con seis intersecciones. En la primera se hicieron 3 experimentos variando los flujos de entrada vehiculares y en la segunda, un solo experimento con giros permitidos en todas las intersecciones lo que hacía la movilidad dentro de la red más dinámica e impredecible. En todos los casos se evaluó el desempeño de un control en lazo abierto o control de tiempos fijos, y el de un control en lazo cerrado que en este caso es representado por el MOGUT. En el 100% de las pruebas el MOGUT resultó como el mejor controlador teniendo en cuenta los índices de desempeño medidos.
- El desarrollo de proyectos cuyo objetivo principal sea mejorar el desempeño de la movilidad vial en el mundo entero es muy importante puesto que se presenta un crecimiento constante en el parque automotor a nivel mundial así como un crecimiento poblacional lo que hace más complicado movilizarse por las calles de las diferentes ciudades. Un documento investigativo realizado por la Universidad de los Andes especula a partir de diferentes estudios económicos y de transporte que para el año 2040 el número de automóviles habrá pasado de 3 millones a 10,4 millones, y el número de motocicletas de 2,4 a 12,9 millones. Con estas cifras y un crecimiento mucho menor en la capacidad

instalada en el país para transporte es necesario pensar en un mayor control del tráfico cada día.

- En el control de tráfico es necesario evolucionar a un tipo de control en tiempo real como el presentado en este proyecto porque el comportamiento del tráfico ya no es tan predecible como en otras épocas. Los recientes cambios tecnológicos y sociales han ocasionado que ya no sea tan fácil crear un patrón común de tráfico por los diferentes horarios laborales que existen. Además eventos especiales como accidentes, manifestaciones y daños en las vías pueden alterar ese patrón o comportamiento en el tráfico y el control en tiempo de real puede brindar una solución además de que permite que dichos inconvenientes sean detectados.

BIBLIOGRAFIA

CASTRO, Antonio y FUSARIO, Ruben. Teleinformática: para ingenieros en sistemas de información. 2 ed. España: Editorial REVERTÉ, S. A., 1999. 396 p. ISBN-84-291-4391. [consultado 17 de Mayo de 2012]. Disponible en Internet: http://books.google.com.co/books?id=ZOZp72uLGMwC&printsec=frontcover&hl=es&source=gbs_ge_summary_r&cad=0

CEBALLOS, Francisco J. Microsoft C# Curso de Programación. México: Alfaomega Grupo Editor, 2007. 833 p. ISBN 978-970-15-1222-7.

COSS, Raúl. Simulación, Un enfoque práctico. México, Editorial LIMUSA, 2003. 157 p. ISBN 968-18-1506-8. [consultado 12 de Enero de 2013]. Disponible en Internet: http://books.google.com.co/books?id=iY6dl3E0FNUC&printsec=frontcover&hl=es&source=gbs_ge_summary_r&cad=0

DUSSICH, Drago. Comunicaciones inalámbricas para el control del tráfico vehicular. En: Revista Colombiana de Telecomunicaciones. Agosto-Octubre, 2010, vol. 17, no. 57, p. 38-41.

GARBER, Nicholas J. y HOEL, Lester A. Ingeniería de Tránsito y Carreteras. 4 ed. Traducido del libro *Traffic and Highway Engineering*. México,D.F.: Thomson Editores, S.A., 2005. 1150 p. [consultado 12 de Enero de 2013]. Disponible en Internet: http://books.google.com.co/books?id=G9zxVrbzctcC&printsec=frontcover&hl=es&source=gbs_ge_summary_r&cad=0

HOFBAUER, Josef. Evolutionary Game Dynamics. En: Bulletin of the American Mathematical Society [en línea], Volume 40, Numer 4. Electronically published, 2003. p. 479-519. [consultado 21 de Junio de 2012]. Disponible en Internet: <http://homepage.univie.ac.at/josef.hofbauer/03bams.pdf>

Introducción al lenguaje C# y .NET Framework [en línea]. Estados Unidos: Microsoft, 2010 [consultado 05 de Diciembre de 2011]. Disponible en Internet: [http://msdn.microsoft.com/es-ES/library/z1zx9t92\(v=vs.100\)](http://msdn.microsoft.com/es-ES/library/z1zx9t92(v=vs.100))

Iteris, Inc. Pico: Compacto, easy-to-use video detection for optimised signal actuation. 2011. [consultado 05 de Diciembre de 2011]. Disponible en Internet: http://www.iteris.com/upload/datasheets/Pico_web.pdf

MOLLER, Rolf. La Alternativa para el Transporte Público Colectivo en Colombia. Cali, Valle.: Programa y Editorial Universidad del Valle, 2004.

OGATA, Katsuhiko. Ingeniería de Control Moderna. 4 ed. Madrid: Pearson Educación, S.A., 2003. 984 p. ISBN 84-205-3678-4.

PTV AG. VISSIM 5.40 – User Manual. Alemania, 2012. 1 archivo de computador.

PTV Group. What Keeps traffic flowing? [en línea]. Alemania. [consultado 05 de Diciembre de 2011]. Disponible en Internet: http://vision-traffic.ptvgroup.com/fileadmin/files_ptvision/Downloads_N/0_General/2_Products/2_PTV_Vissim/EN-UK_PTV_Vissim_Brochure.pdf

SCATS [en línea]. Australia: Transport, roads and maritime services, NSW Government [consultado 05 de Diciembre de 2011]. Disponible en Internet: http://www.scats.com.au/product_base_packg_compnts.html

SHOHAM, Yoav. Agent-oriented Programming. En: Artificial Intelligence 60. Standford: Elsevier, 1993. p. 51-92. [consultado 24 de Noviembre de 2011]. Disponible en Internet: <http://123seminaronly.com/Seminar-Reports/050/62112045-Agent-Oriented-Programming.pdf>

SPIEGELMAN, Clifford H. PARK, Eun S. RILETT, Laurence R. Transportation: Statistics and Microsimulation. Boca Ratón, FL.: Chapman & Hall/CRC Press, 2011. 355 p. ISBN 978-1-4398-0023-2.

Soluciones en Tiempo Real Limitada. Central de Tráfico IMATIC – RTS: Manual de Usuario. [consultado 24 de Noviembre de 2011]. Disponible en internet: http://www.rts.com.co/Central/Manual_Central_V2.pdf

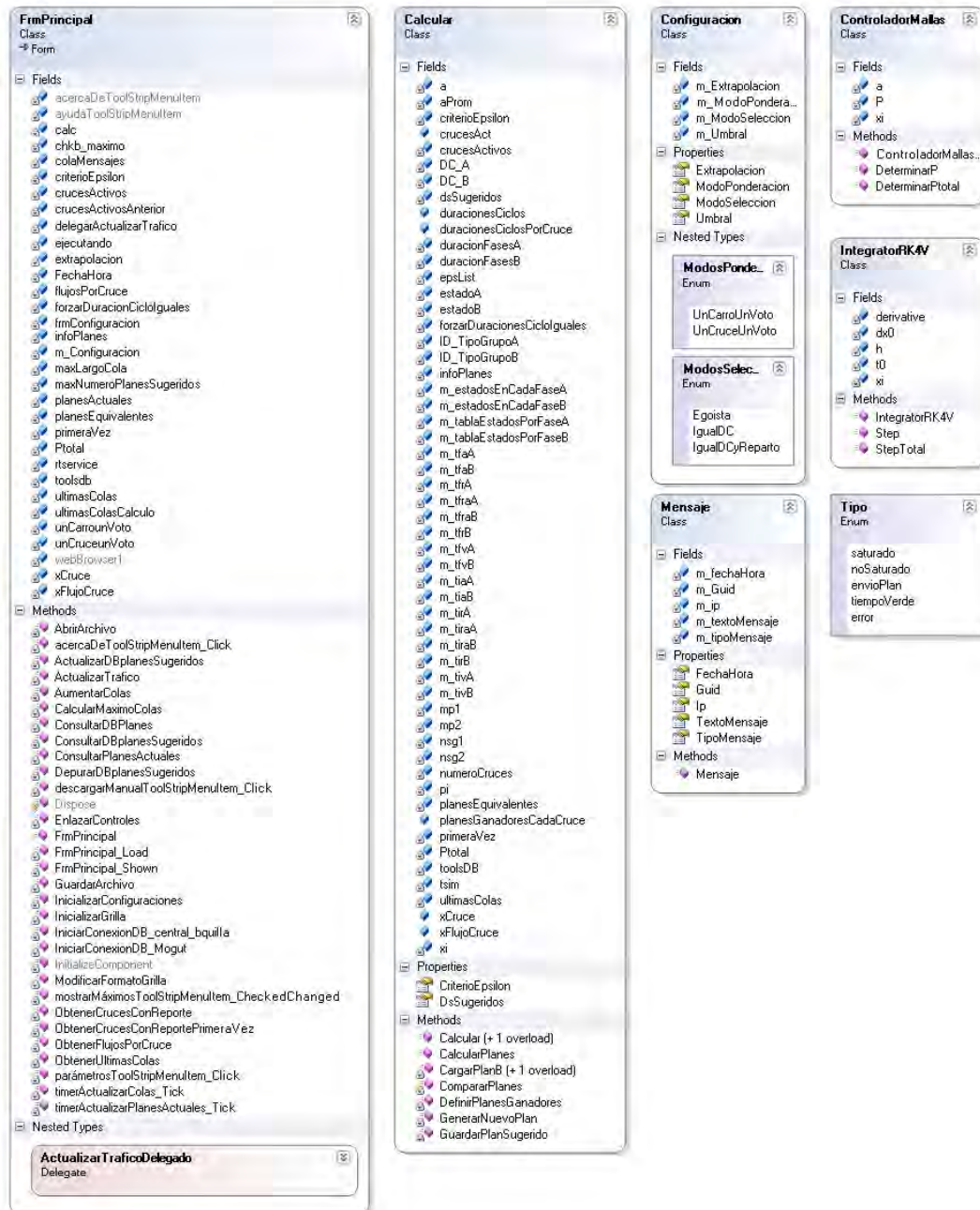
STIDHAM, Shaler Jr. Optimal Design of Queueing Systems. Boca Raton, FL.: Chapman & Hall/CRC Press, 2009.

VANCELLS, Joan. Algoritmos y Programas. Editorial UOC. 21 p. P01/79008/00053. [consultado 12 de Enero de 2013]. Disponible en Internet: <http://books.google.com.co/books?id=O55qXjsDjnYC&printsec=frontcover&hl=es>

What is Scoot? [en línea]. Peek Traffic Limited, Siemens Traffic Controls y TRL Limited [consultado 05 de Diciembre de 2011]. Disponible en Internet: <http://www.scoot-utc.com/WhatIsSCOOT.php?menu=Overview>

ANEXOS

Anexo A. Diagrama de clases MOGUT



Anexo B. Datasheet Tarjeta de adquisición de datos NI USB-6509

High-Density Industrial Digital I/O for USB – 96-Channel, 5 V TTL/CMOS



Portable digital I/O device

Built-in, selectable pull-up/pull-down resistors

96 bidirectional I/O channels (5 V TTL/CMOS)

Direct connection to 5 V logic devices and most solid-state relays

High-current drive (up to 24 sink or source)

Optional power supply for additional sourcing current

Low-cost solution with superior features for data acquisition, manufacturing test, and industrial control applications

Per-channel selectable debounce filters for digital input lines

Overview

The NI USB-6509 is a portable, industrial data acquisition device with 96 channels of digital I/O. It is compatible with TTL, CMOS, and 5 V digital logic levels. You can individually configure each port of eight lines for input or output. You can drive external digital devices, such as solid-state relays (SSRs), with current up to 24 mA per line using power available over USB. In bus-powered mode, the USB-6509 can source up to 215 mA total across all digital output lines. Use the optional 12 VDC power supply for additional sourcing current. A USB-6509 is ideal for general-purpose data acquisition applications as well as industrial control and automated manufacturing test. With high-current drive, you can connect the digital I/O device directly to a wide array of 5 V electronic devices, sensors, and actuators. Industrial digital I/O devices are designed to incorporate the latest hardware technologies for applications requiring ease of use, high reliability, and performance. The USB-6509 takes advantage of NI-DAQmx measurement services software to speed up application development with features such as the DAQ Assistant, automatic code generation, and high-performance multithreaded streaming technology.

Requirements and Compatibility

OS Information	Driver Information	Software Compatibility
Windows 2000/XP	NI-DAQmx	ANSI C/C++
Windows Vista x64/x86		LabVIEW
		LabVIEW SignalExpress
		LabWindows/CVI
		Measurement Studio
		Visual Studio .NET

Application and Technology

Hardware

Industrial Feature Set.

NI industrial digital I/O devices offer a set of high-reliability features designed to automate even the most demanding applications.

- Programmable power-up states provide safe operation when connected to pumps/valves/motors/relays
- Digital I/O watchdogs detect computer or application crashes and ensure safe recovery

- Change-detection triggers your application and performs a read operation after a digital event with minimal processor usage
- Programmable input filters eliminate glitches/spikes and remove noise

Glitch-Free Startup with Programmable Power-Up States.

Using programmable power-up states, you can configure the initial USB-6509 output states in software to ensure consistent, reliable operation when connected to industrial actuators such as pumps, valves, motors, and relays. A USB-6509 holds these output states after receiving power, so your computer can boot and your software application can begin running. Programmable power-up states are glitch-free, meaning the outputs never go through an incorrect state during power up.

You can configure each digital line as high-impedance-input, high-output, or low-output. A USB-6509 stores the settings in onboard nonvolatile memory and implements the power-up states instantaneously after power is applied to the device.

Detect and Recover with Digital I/O Watchdogs.

Digital I/O watchdogs are an innovative technology that can detect a variety of fault conditions such as an application crash and automatically respond by setting the outputs to a user-configured safe state. Watchdogs are important whenever the module is connected to actuators such as pumps, valves, motors, and relays. A USB-6509 monitors the software application; if the application fails to respond within a preset time limit, the device automatically sets the output lines to a user-defined safe state. A USB-6509 remains in the watchdog state until the watchdog timer is disarmed, the device is reset, or the computer is restarted.

Trigger Your Application with Change Detection.

With change detection, you can automatically trigger your software application to perform a digital read operation upon a digital change of state. A digital change of state is defined as the rising edge (0 to 1 transition) or falling edge (1 to 0 transition) on one or more digital lines. Using change detection, you can monitor for digital events with minimal processor usage. No polling is necessary because the digital I/O module generates an interrupt to automatically wake up your application. Using NI-DAQmx software technology, the USB-6509 notifies the software application when it detects an event, causing the application to automatically perform a read operation. To minimize the effects of noisy input lines, use programmable input filters in combination with change detection to eliminate spurious change-detection events caused by noise or glitches.

Eliminate Noise with Programmable Input Filters.

Programmable input filters remove noise, glitches, and spikes on inputs as well as provide debouncing for digital switches and relays. These features are important for applications in industrial environments to prevent false readings caused by noise. You can configure the programmable input filter for each digital line by setting the filter time. The USB-6509 blocks any digital noise, glitch, or spike that is shorter than half of the specified filter time, preventing invalid readings and false triggers for change-detection events.

Recommended Software.

National Instruments measurement services software, built around NI-DAQmx driver software, includes intuitive application programming interfaces, configuration tools, I/O assistants, and other tools designed to reduce system setup, configuration, and development time. National Instruments recommends using the latest version of NI-DAQmx driver software for application development in NI LabVIEW, LabWindows™/CVI, and Measurement Studio. To obtain the latest version of NI-DAQmx, visit ni.com/support/daq/versions. NI measurement services software speeds up your development with features including the following:

- A guide to create fast and accurate measurements with no programming using the DAQ Assistant
- Free LabVIEW SignalExpress LE data-logging software
- Automatic code generation to create your application in LabVIEW; LabWindows/CVI; LabVIEW SignalExpress; and Visual Studio .NET, ANSI C/C++, C#, or Visual Basic 6.0
- Multithreaded technology for 1,000 times performance improvements over basic DAQ driver software
- More than 3,000 free software downloads to jump-start your project at ni.com/zone

The USB-6509 is compatible with the following versions (or later) of NI application software: LabVIEW, LabWindows/CVI, or Measurement Studio versions 7.x; or LabVIEW SignalExpress 2.5. The USB-6509 is not compatible with the Traditional NI-DAQ (Legacy) driver.

Digital	I/O	Cables	and	Accessories
The 100-pin high-density SCSI connector on the USB-6509 interfaces to 100-pin ribbon cables or shielded cables. For low-cost unshielded connectivity, use the R1005050 ribbon cable with two CB-50LP or CB-50 connector blocks (a CB-100 kit). For shielded connectivity, use the SH100-100-F shielded digital I/O cable with the SCB-100 connector block.				
R1005050	– Unshielded ribbon cable that terminates with two 50-pin IDC connectors.			
0.5 m182762-0R5
1 m182762-01
2 m182762-02
SH100-100-F	– Shielded 100-conductor cable that terminates with a 100-pin 0.050 series D-type connector that attaches to 100-pin accessories.			
1 m185095-01
2 m185095-02
SCB-100	– Shielded I/O connector block with screw terminals and a general breadboard area for 100-pin digital I/O devices.			
SCB-100776990-01
CB-50	– Unshielded I/O connector block with DIN-rail mounting and screw terminals. Includes 50-pin header for direct connection to 50-pin cables.			
CB-50776164-90
CB-50LP	– Unshielded termination board with 50 screw terminals. Includes a 50-pin header for direct connection to 50-pin cables.			
CB-50LP777101-01
CB-100 Connector Kit	– Includes two CB-50 I/O connector blocks and a 1 m R1005050 ribbon cable.			
CB-100 with 1 m R1005050 cable777812-01
PCB Mounting Connectors	– PCB connectors for use in building custom accessories that connect to 100-conductor shielded and ribbon cables.			
100-pin, female, right-angle mounting777778-01
100-pin, female, vertical mounting777779-01
12 VDC Power Supply	– Optional 12 VDC, 1.0 A power supply and cable for up to 24 mA current drive per channel.			
12 VDC power supply780308-0X



R1005050



SH100-100-F



SCB-100



CB-50



CB-50LP



CB-100



PCB Mounting Connectors

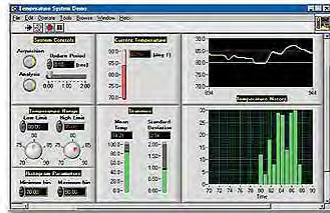
Ordering Information

For a complete list of accessories, visit the product page on ni.com.

Products	Part Number	Recommended Accessories	Part Number
NI USB-6509			
NI USB-6509	779975-	Cable: Shielded - SH100-	185095-
Each NI USB-6509 requires: 1	01	100-F Cable (2m)	02
Cable, 1 Connector Block		**Also available:	
		Unshielded	
		Connector Block: Screw	776990-
		Terminals - SCB-100	01

Software Recommendations

LabVIEW Professional Development System for Windows



Advanced software tools for large project development

Automatic code generation using DAQ Assistant and Instrument I/O Assistant

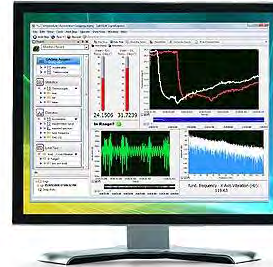
Tight integration with a wide range of hardware

Advanced measurement analysis and digital signal processing

Open connectivity with DLLs, ActiveX, and .NET objects

Capability to build DLLs, executables, and MSI installers

NI LabVIEW SignalExpress for Windows



Quickly configure projects without programming

Control over 400 PC-based and stand-alone instruments

Log data from more than 250 data acquisition devices

Perform basic signal processing, analysis, and file I/O

Scale your application with automatic LabVIEW code generation

Create custom reports or easily export data to LabVIEW, DIAdem or Microsoft Excel

NI LabWindows™/CVI for Windows

Real-time advanced 2D and graphs

NI Measurement Studio Enterprise Edition

Support for Microsoft Visual Studio .NET 2010/2008/2005

charts



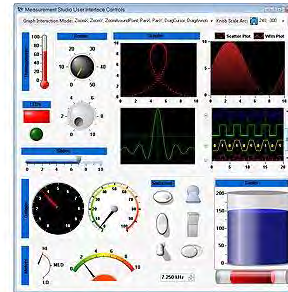
Complete hardware compatibility with IVI, VISA, DAQ, GPIB, and serial

Analysis tools for array manipulation, signal processing, statistics, and curve fitting

Simplified cross-platform communication with network variables

Measurement Studio .NET tools (included in LabWindows/CVI Full only)

The mark LabWindows is used under a license from Microsoft Corporation.



Customizable Windows Forms and Web Forms controls for test and measurement user interface design

Hardware integration support with data acquisition and instrument control libraries

Automatic code generation with data acquisition, instrument control, and parameter assistants

LabWindows™/CVI ANSI C development environment

Advanced analysis and cross-platform communication libraries

Support and Services

System Assurance Programs.

NI system assurance programs are designed to make it even easier for you to own an NI system. These programs include configuration and deployment services for your NI PXI, CompactRIO, or Compact FieldPoint

system. The NI Basic System Assurance Program provides a simple integration test and ensures that your system is delivered completely assembled in one box. When you configure your system with the NI Standard System Assurance Program, you can select from available NI system driver sets and application development environments to create customized, reorderable software configurations. Your system arrives fully assembled and tested in one box with your software preinstalled. When you order your system with the standard program, you also receive system-specific documentation including a bill of materials, an integration test report, a recommended maintenance plan, and frequently asked question documents. Finally, the standard program reduces the total cost of owning an NI system by providing three years of warranty coverage and calibration service. Use the online product advisors at ni.com/advisor to find a system assurance program to meet your needs.

Calibration.

NI measurement hardware is calibrated to ensure measurement accuracy and verify that the device meets its published specifications. To ensure the ongoing accuracy of your measurement hardware, NI offers basic or detailed recalibration service that provides ongoing ISO 9001 audit compliance and confidence in your measurements. To learn more about NI calibration services or to locate a qualified service center near you, contact your local sales office or visit ni.com/calibration.

Technical Support

Get answers to your technical questions using the following National Instruments resources.

- **Support** - Visit ni.com/support to access the NI KnowledgeBase, example programs, and tutorials or to contact our applications engineers who are located in NI sales offices around the world and speak the local language.
- **Discussion Forums** - Visit forums.ni.com for a diverse set of discussion boards on topics you care about.
- **Online Community** - Visit community.ni.com to find, contribute, or collaborate on customer-contributed technical content with users like you.

Repair

While you may never need your hardware repaired, NI understands that unexpected events may lead to necessary repairs. NI offers repair services performed by highly trained technicians who quickly return your device with the guarantee that it will perform to factory specifications. For more information, visit ni.com/repair.

Training and Certifications

The NI training and certification program delivers the fastest, most certain route to increased proficiency and productivity using NI software and hardware. Training builds the skills to more efficiently develop robust, maintainable applications, while certification validates your knowledge and ability.

- **Classroom training in cities worldwide** - the most comprehensive hands-on training taught by engineers.
- **On-site training at your facility** - an excellent option to train multiple employees at the same time.
- **Online instructor-led training** - lower-cost, remote training if classroom or on-site courses are not possible.

- **Course kits** - lowest-cost, self-paced training that you can use as reference guides.
 - **Training memberships** and training credits - to buy now and schedule training later.
- Visit ni.com/training for more information.

Extended Warranty

NI offers options for extending the standard product warranty to meet the life-cycle requirements of your project. In addition, because NI understands that your requirements may change, the extended warranty is flexible in length and easily renewed. For more information, visit ni.com/warranty.

OEM

NI offers design-in consulting and product integration assistance if you need NI products for OEM applications. For information about special pricing and services for OEM customers, visit ni.com/oem.

Alliance

Our Professional Services Team is comprised of NI applications engineers, NI Consulting Services, and a worldwide National Instruments Alliance Partner program of more than 700 independent consultants and integrators. Services range from start-up assistance to turnkey system integration. Visit ni.com/alliance.

Detailed Specifications

These specifications are typical at 25 °C, unless otherwise state.

Number of channels	96 input/output
Compatibility	TTL/CMOS, single-ended GND referenced
Power-on state	Configured as inputs, pull-up/pull- down (selectable on the back panel switch)
Pull-up/pull-down resistor	100 kΩ ±5%
Data transfers	Programmed I/O

Digital Input Characteristics

Level	Min	Max
Input voltage	0 V	5.0 V
Input low voltage	—	0.8 V
Input high voltage	2.0 V	—
Input low current ($V_{in} = 0$ V)	—	−91.0 μ A
Input high current ($V_{in} = 5$ V)	—	91.0 μ A

Digital Output Characteristics

Level	Min	Typ	Max
Low-level output current (I_{OL})	—	—	24 mA
High-level output current (I_{OH})	—	—	−24 mA
Output low voltage (V_{OL}), at 100 μ A	—	0.0 V	0.2 V
at 2 mA	—	0.1 V	—
at 12 mA	—	0.5 V	—
at 24 mA	—	1.0 V	1.4 V
Output high voltage (V_{OH}), at −100 μ A	4.3 V	5.0 V	—
at −2 mA	—	4.9 V	—
at −12 mA	—	4.5 V	—
at −24 mA	2.8 V	4.0 V	—

The total current sinking/sourcing from one port cannot exceed 100 mA.
 Total current sourced by all DO lines simultaneously should not exceed 215 mA for bus-powered configuration.
 If more sourcing current is required, use an optional external power supply
 Total current sourced by all DO lines simultaneously should not exceed 215 mA for bus-powered configuration.
 If more sourcing current is required, use an optional external power supply

+5V power available at I/O connector (pins 49 and 99)	+4.1 to +5.2 V; 215 mA, maximum (Bus-powered); 0.5 A, maximum (with external power supply)
---	--

Note The I/O connector power has a fuse for overcurrent protection. This fuse is not customer replaceable. If the fuse is blown, return the device to NI for repair.

Bus Interface

USB	USB 2.0 Hi-Speed or full-speed
-----	--------------------------------

Power Requirement

USB	
Input voltage	4.5 to 5.25 V in configured state
No load current	150 mA, max
Maximum load current	500 mA, max
Suspend current	250 μ A, typical (no external supply)
External DC supply	
Input voltage range	+12 V \pm 20%
Maximum input current	1 A
Power input connector	DC input jack with threaded locking ring, 0.08 in. (2 mm) center pin
Power input mating connector	Switchcraft S760K

Physical

Dimensions	17.78 \times 10.30 \times 3.34 cm (7.0 \times 4.1 \times 1.3 in.)
I/O connector	100-pin female 0.050 series SCSI
Weight	Approx. 239 g (8.4 oz)

Safety

If you need to clean the module, wipe it with a dry towel.

Safety Standards

This product is designed to meet the requirements of the following standards of safety for electrical equipment for measurement, control, and laboratory use:

IEC 61010-1, EN 61010-1
 UL 61010-1, CSA 61010-1

Note For UL and other safety certifications, refer to the product label or the *Online Product Certification* section.

Hazardous Locations

This device is not certified for use in hazardous locations.

Environmental

The NI USB-6509 device is intended for indoor use only.

Operating temperature (IEC 60068-2-1 and IEC 60068-2-2)	0 to 55 °C
---	------------

Operating humidity (IEC 60068-2-56)	10 to 90% RH, noncondensing
-------------------------------------	-----------------------------

Maximum altitude	2,000 m (at 25 °C ambient temperature)
------------------	--

Storage temperature (IEC 60068-2-1 and IEC 60068-2-2)	– 40 to 85 °C
---	---------------

Storage humidity (IEC 60068-2-56)	5 to 90% RH, noncondensing
-----------------------------------	----------------------------

Pollution Degree (IEC 60664)	2
------------------------------	---

Electromagnetic Compatibility

This product is designed to meet the requirements of the following standards of EMC for electrical equipment for measurement, control, and laboratory use:

EN 61326 EMC requirements; Minimum Immunity

EN 55011 Emissions; Group 1, Class A

CE, C-Tick, ICES, and FCC Part 15 Emissions; Class A

Note For EMC compliance, operate this device with double-shielded cables.

CE Compliance

This product meets the essential requirements of applicable European Directives, as amended for CE marking, as follows:

2006/95/EC; Low-Voltage Directive (safety)

2004/108/EC; Electromagnetic Compatibility Directive (EMC)

Note For the standards applied to assess the EMC of this product, refer to the *Online Product Certification* section.

Online Product Certification

Refer to the product Declaration of Conformity (DoC) for additional regulatory compliance information. To obtain product certifications and the DoC for this product, visit ni.com/certification, search by module number or product line, and click the appropriate link in the Certification column.

Environmental Management

National Instruments is committed to designing and manufacturing products in an environmentally responsible manner. NI recognizes that eliminating certain hazardous substances from our products is beneficial not only to the environment but also to NI customers.

For additional environmental information, refer to the *NI and the Environment* Web page at ni.com/environment. This page contains the environmental regulations and directives with which NI complies, as well as other environmental information not included in this document.

Waste Electrical and Electronic Equipment (WEEE)

EU Customers At the end of their life cycle, all products *must* be sent to a WEEE recycling center. For more information about WEEE recycling centers and National Instruments WEEE initiatives, visit ni.com/environment/weee.htm.

电子信息产品污染控制管理办法（中国 RoHS）



中国客户 National Instruments 符合中国电子信息产品中限制使用某些有害物质指令。关于 National Instruments 中国 RoHS 合规性信息，请登录 ni.com/environment/。
(For Information about China RoHS compliance, go to ni.com/environment/.)

Pinouts/Front Panel Connections

P2.7	1	51	P8.7
P5.7	2	52	P11.7
P2.6	3	53	P8.6
P5.6	4	54	P11.6
P2.5	5	55	P8.5
P5.5	6	56	P11.5
P2.4	7	57	P8.4
P5.4	8	58	P11.4
P2.3	9	59	P8.3
P5.3	10	60	P11.3
P2.2	11	61	P8.2
P5.2	12	62	P11.2
P2.1	13	63	P8.1
P5.1	14	64	P11.1
P2.0	15	65	P8.0
P5.0	16	66	P11.0
P1.7	17	67	P7.7
P4.7	18	68	P10.7
P1.6	19	69	P7.6
P4.6	20	70	P10.6
P1.5	21	71	P7.5
P4.5	22	72	P10.5
P1.4	23	73	P7.4
P4.4	24	74	P10.4
P1.3	25	75	P7.3
P4.3	26	76	P10.3
P1.2	27	77	P7.2
P4.2	28	78	P10.2
P1.1	29	79	P7.1
P4.1	30	80	P10.1
P1.0	31	81	P7.0
P4.0	32	82	P10.0
P0.7	33	83	P6.7
P3.7	34	84	P9.7
P0.6	35	85	P6.6
P3.6	36	86	P9.6
P0.5	37	87	P6.5
P3.5	38	88	P9.5
P0.4	39	89	P6.4
P3.4	40	90	P9.4
P0.3	41	91	P6.3
P3.3	42	92	P9.3
P0.2	43	93	P6.2
P3.2	44	94	P9.2
P0.1	45	95	P6.1
P3.1	46	96	P9.1
P0.0	47	97	P6.0
P3.0	48	98	P9.0
+5 V	49	99	+5 V
GND	50	100	GND

Pin Assignments for the NI USB-6509 using the SH100-100-F

Positions 1 through 50

P2.7	1	2	P5.7
P2.6	3	4	P5.6
P2.5	5	6	P5.5
P2.4	7	8	P5.4
P2.3	9	10	P5.3
P2.2	11	12	P5.2
P2.1	13	14	P5.1
P2.0	15	16	P5.0
P1.7	17	18	P4.7
P1.6	19	20	P4.6
P1.5	21	22	P4.5
P1.4	23	24	P4.4
P1.3	25	26	P4.3
P1.2	27	28	P4.2
P1.1	29	30	P4.1
P1.0	31	32	P4.0
P0.7	33	34	P3.7
P0.6	35	36	P3.6
P0.5	37	38	P3.5
P0.4	39	40	P3.4
P0.3	41	42	P3.3
P0.2	43	44	P3.2
P0.1	45	46	P3.1
P0.0	47	48	P3.0
+5 V	49	50	GND

Positions 51 through 100

P8.7	1	2	P11.7
P8.6	3	4	P11.6
P8.5	5	6	P11.5
P8.4	7	8	P11.4
P8.3	9	10	P11.3
P8.2	11	12	P11.2
P8.1	13	14	P11.1
P8.0	15	16	P11.0
P7.7	17	18	P10.7
P7.6	19	20	P10.6
P7.5	21	22	P10.5
P7.4	23	24	P10.4
P7.3	25	26	P10.3
P7.2	27	28	P10.2
P7.1	29	30	P10.1
P7.0	31	32	P10.0
P6.7	33	34	P9.7
P6.6	35	36	P9.6
P6.5	37	38	P9.5
P6.4	39	40	P9.4
P6.3	41	42	P9.3
P6.2	43	44	P9.2
P6.1	45	46	P9.1
P6.0	47	48	P9.0
+5 V	49	50	GND

Anexo C. Manual de usuario MOGUT



2/F/01/02



NIT. No. 800.195.522-1

MÓDULO OPTIMIZADOR MULTINODAL PARA MALLAS DE TRÁFICO URBANO MOGUT (MULTINODAL OPTIMIZATION FOR GRIDS OF URBAN TRAFFIC)

CENTRAL DE TRÁFICO IMATIC - RTS

MANUAL DE USUARIO - SERVIDOR



2/F/01/02



NIT. No. 800.195.522-1

DESCRIPCIÓN

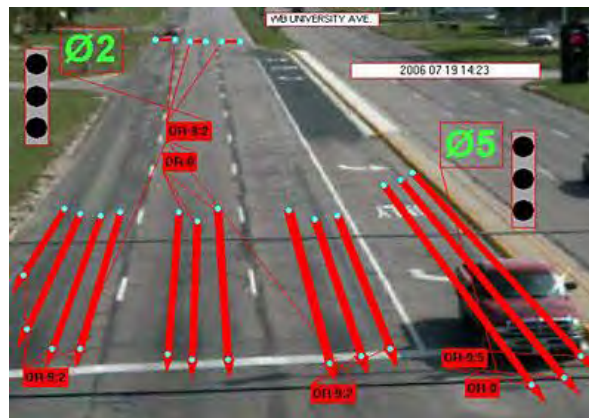
El Módulo de Gestión para Control de Corredores y Mallas es un sistema adaptable (auto-regulable) de distribución de tiempos de paso en una intersección Semaforizada, basándose en información suministrada por sistemas de video-detección (cámaras especializadas) y un algoritmo de inteligencia computacional.

REQUERIMIENTOS

SENSORES:

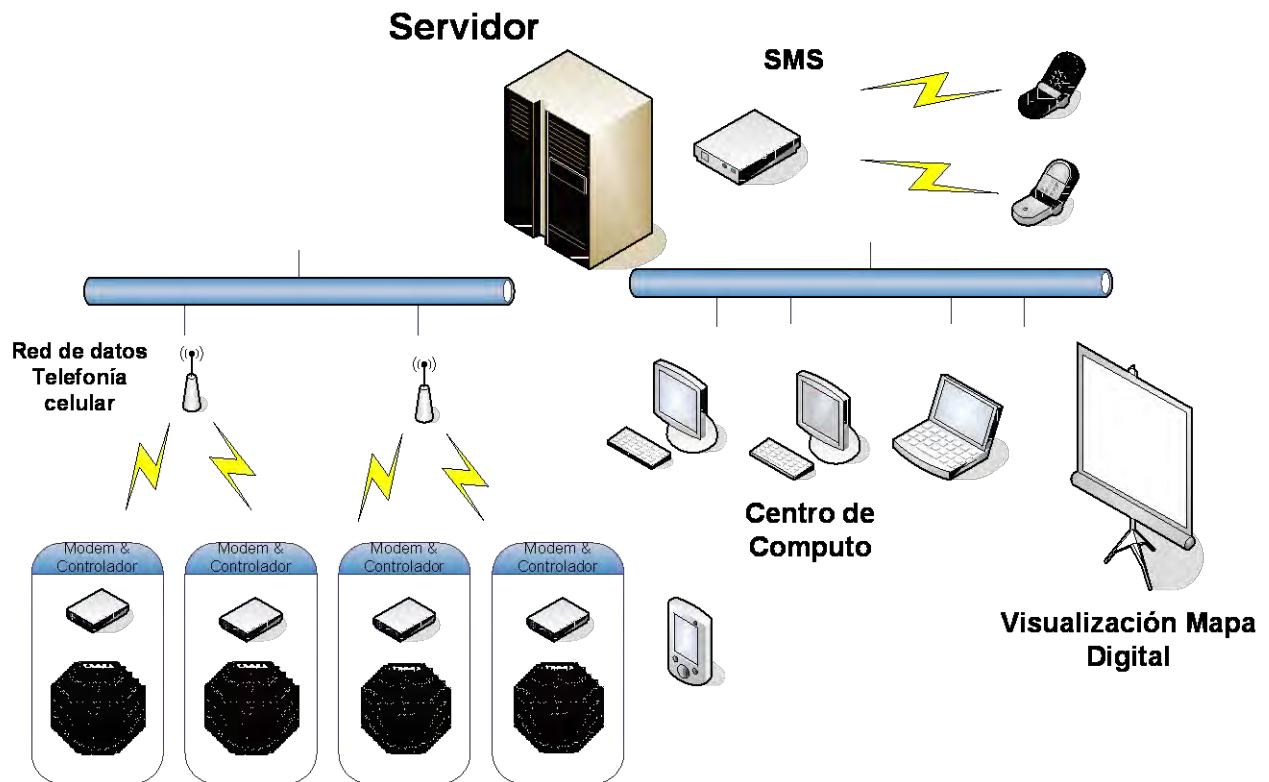
Instalación de sensores en cada uno de los flujos que hagan parte de la malla.

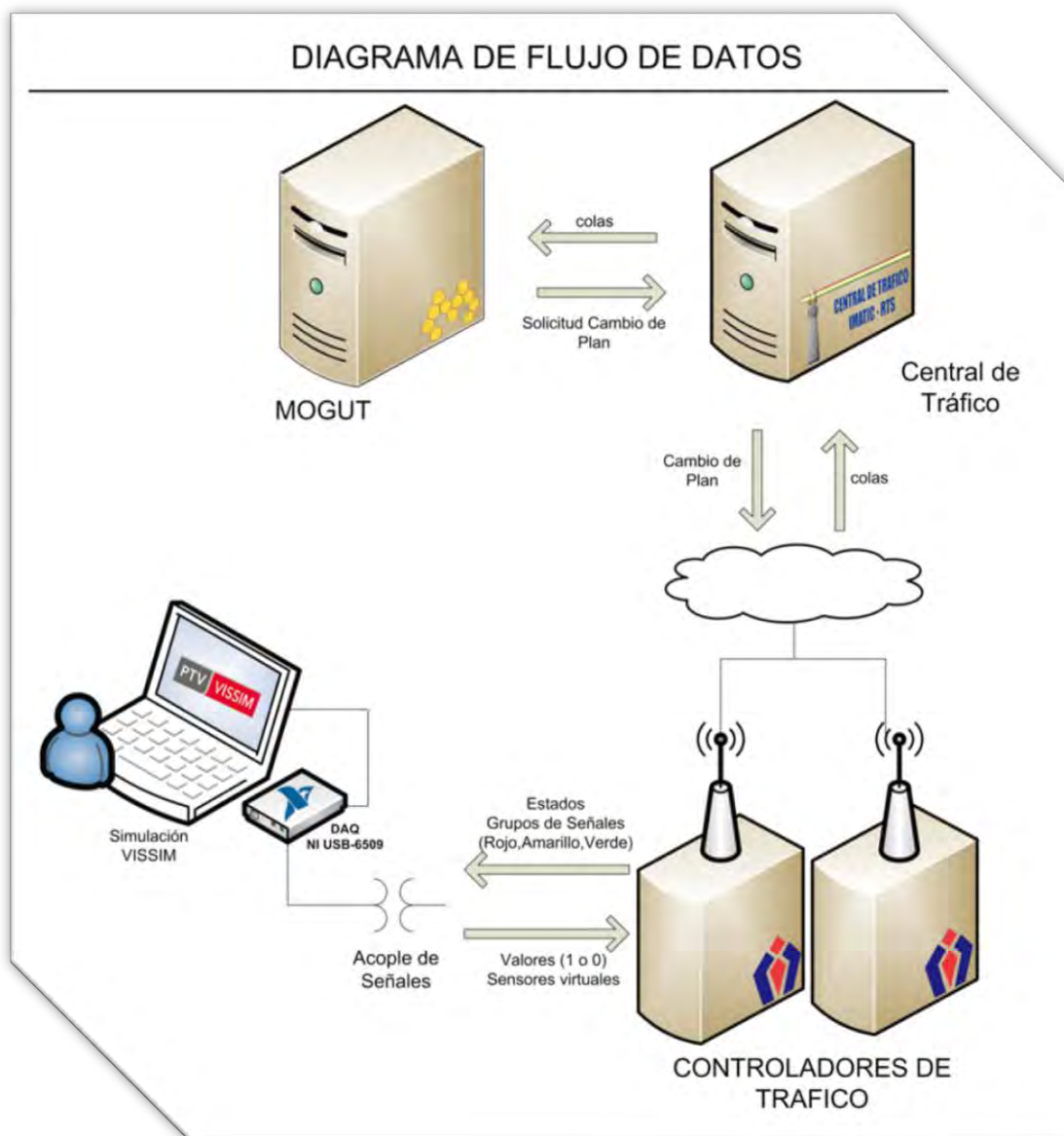
Independiente de la tecnología empleada en la detección, radar, espiras magnéticas, video, etc., la mayoría de los sensores disponibles en el mercado entregan una salida por cada detector en la forma de un contacto seco de relé, lo que permite su implementación de forma transparente en el sistema.



COMUNICACIÓN

El sistema hace uso intensivo de transmisiones de datos para conocer los valores de los contadores en campo y enviar los ajustes







2/F/01/02



NIT. No. 800.195.522-1

METODO DE OPTIMIZACIÓN

- Uso del método de algoritmo de optimización basado en inteligencia computacional
- Busca la minimización de la cola (global).
- Afecta en forma sucesiva tres parámetros:
 - Repartición.
 - Desfase.
 - Duración del ciclo.

DESCRIPCION DE LA INTERFAZ

MOGUT - Central de Trafico Imatic-RTS.

Configuración Ayuda

Monitoreo Colas y Planes Actuales | Monitoreo Planes Sugeridos

Registro de eventos

Plan a Enviar	2013-02-12 11:56:56
cruce = 8004 Estructura = 2 Plan = 4 indice = 46	
Plan a Enviar	2013-02-12 11:56:56
cruce = 8003 Estructura = 2 Plan = 4 indice = 46	
Plan a Enviar	2013-02-12 11:56:56
cruce = 8002 Estructura = 2 Plan = 4 indice = 46	
Plan a Enviar	2013-02-12 11:56:56
cruce = 8001 Estructura = 2 Plan = 4 indice = 46	
Tiempo de Verde para la Red	2013-02-12 11:56:56
140	
Ocupacion Flujo	2013-02-12 11:56:56
cruce = 8004 flujo = 8 indice Desempeno = 0 saturado : NO	
Ocupacion Flujo	2013-02-12 11:56:56
cruce = 8004 flujo = 7 indice Desempeno = 0 saturado : NO	
Ocupacion Flujo	2013-02-12 11:56:56
cruce = 8003 flujo = 6 indice Desempeno = 0 saturado : NO	
Ocupacion Flujo	2013-02-12 11:56:56
cruce = 8003 flujo = 5 indice Desempeno = 0 saturado : NO	
Ocupacion Flujo	2013-02-12 11:56:56
cruce = 8002 flujo = 4 indice Desempeno = 0 saturado : NO	
Ocupacion Flujo	2013-02-12 11:56:56
cruce = 8002 flujo = 3 indice Desempeno = 0 saturado : NO	
Ocupacion Flujo	2013-02-12 11:56:56
cruce = 8001 flujo = 2 indice Desempeno = 0 saturado : NO	
Ocupacion Flujo	2013-02-12 11:56:56
cruce = 8001 flujo = 1 indice Desempeno = 0 saturado : NO	
Plan a Enviar	2013-02-12 11:56:27
cruce = 8004 Estructura = 2 Plan = 4 indice = 46	

Colas

Drag a column header here to group by that column.

codcruce	numeroflujo	cols	saturado	datecreated
8001	1	0	0	2013-02-12 11:56:45
8001	2	0	0	2013-02-12 11:56:45
8002	3	0	0	2013-02-12 11:56:45
8002	4	0	0	2013-02-12 11:56:45
8003	5	0	0	2013-02-12 11:56:45
8003	6	0	0	2013-02-12 11:56:45
8004	7	0	0	2013-02-12 11:56:45
8004	8	0	0	2013-02-12 11:56:45
8001	1	0	0	2013-02-12 11:56:34
8001	2	0	0	2013-02-12 11:56:34
8002	3	0	0	2013-02-12 11:56:34
8002	4	0	0	2013-02-12 11:56:34
8003	5	0	0	2013-02-12 11:56:34
8003	6	0	0	2013-02-12 11:56:34
8004	7	0	0	2013-02-12 11:56:34
8004	8	0	0	2013-02-12 11:56:34
8001	1	0	0	2013-02-12 11:56:24
8001	2	0	0	2013-02-12 11:56:24
8002	3	0	0	2013-02-12 11:56:24
8002	4	0	0	2013-02-12 11:56:24
8003	5	0	0	2013-02-12 11:56:24
8003	6	0	0	2013-02-12 11:56:24
8004	7	0	0	2013-02-12 11:56:24

Planes Actuales

Drag a column header here to group by that column.

codigo	numeroestructura	numeroplan	duracionciclo
8001	1	6	60
8002	1	6	60
8003	1	6	60
8004	1	6	60

La aplicación está diseñada para demandar un mínimo de interacción con el usuario, a parte de las configuraciones iniciales, la principal función de la interfaz es permitir la supervisión de la operación en la cual se imprime continuamente los valores calculados y los resultados de la interacción con la central de tráfico e indirectamente con los controladores.

CONFIGURACIÓN DEL OPTIMIZADOR



Una vez configurados adecuadamente los planes de señales, el usuario sólo debe configurar pocos parámetros para iniciar el proceso de optimización.

Dada la intención de que el proceso posea un alto nivel de automatismo y adaptabilidad a los cambios en los volúmenes de tráfico.

Modo de Selección

Para cada cruce y condición de tráfico existe un plan ideal que cumpla con las mejores condiciones de duración de ciclo, que se determina a partir del número de vehículos a la espera, y el reparto que se determina a partir de las proporciones relativas de vehículos a la espera en cada flujo del cruce.

Las condiciones en cada cruce son diferentes y por tanto el plan con el mejor índice de desempeño, es decir, el de mayor conveniencia para arbitrar la dinámica instantánea será diferente en cada cruce, por ello se han establecido 3 modos de selección de los planes que gobernarán cada cruce por separado o la malla en general:



1. Egoísta: Cada cruce escoge el plan que lo gobernará sin considerar las selecciones realizadas por los demás cruces; es ideal para cruces que poseen distancias considerables entre sí (superiores a 600m) o con significativas influencias sobre su dinámica i.e. pasos peatonales de alto flujo, pendientes significativas que dificulten el arranque, entre otras.
2. Comparten DC: Los cruces realizan un proceso de votación en el que ponderan y deciden que duración de ciclo es mejor para la mayoría de ellos, pero escogen de forma independiente el reparto.
3. Comparten DC y Reparto: Los cruces realizan un proceso de votación en el que ponderan y deciden qué valores de duración de ciclo y reparto son más convenientes para la red, para ello es necesario que los planes cargados en el equipo tengan identificaciones equivalentes (ver Configuración de Planes Para Uso de Mallas).

Modo de ponderación

Al momento de la votación existen dos mecanismos para ponderar la selección de los planes que afectaran a la red en general, estos sólo se aplican para modos de selección diferentes del Egoísta.

1. Un auto un voto: En la búsqueda de un optimo social, se selecciona este mecanismo de tal manera que las decisiones tengan en cuenta a cuantos usuario de la red benefician. Una selección más cercana al optimo social, consistiría en beneficiar a los vehículos que transporten más personas (ver Módulo de Priorización Imatic-RTS, en nuestro sitio web).
2. Un cruce un voto: Aquí se ponderan las decisiones sin tener en cuenta cuantos vehículos están en las colas, es útil, si los cruces en la malla difieren mucho en sus capacidades, i.e. ancho de la vía, longitud máxima de la cola, entre otros.



2/F/01/02



NIT. No. 800.195.522-1

Umbral

Determina la tolerancia del valor obtenido en la métrica en el cual dos planes se consideran iguales, su función es determinar si un nuevo plan se agregara a la lista de planes sugeridos.

Aunque el cálculo de dicho valor requiere un entendimiento pleno de la comparación de los planes en el espacio híbrido, puede sintonizarse dependiendo de la frecuencia con la que se actualizaran los planes en el controlador en campo, a mayor frecuencia de actualización, mayor generación de planes y por tanto menor el umbral.

Extrapolación

La mayoría de los sensores existentes en el mercado registran sólo una fracción de la cola, la más cercana a la intersección, dejando siempre una cantidad de vehículos sin contar. Esto se debe principalmente a las limitaciones ópticas de los videodetectores, o a los cotos de cableado y canalización de sensores en el suelo que deberían ser instalados muy lejos del equipo controlador.

El algoritmo incrementa artificialmente el valor de la cola (la extrapola) con base en este parámetro y si se dan las siguientes condiciones:

1. El semáforo debe encontrarse en rojo.
2. El valor de extrapolación debe ser mayor a 0.
3. Los contadores positivos, los que incrementan el valor de la cola, al estar ubicados en su entrada (lejos del cruce), deben estar presionados todos a la vez, indicando que la cola está llena hasta donde es posible medirla.

El valor por defecto de esta extrapolación es 0,5 que representa la cantidad media de vehículos que acceden a un carril por segundo en hora pico, pero debe ajustarse si los aforos resultan ser mucho mayores que los valores registrados en el sistema.

Planes Sugeridos

El algoritmo calcula el plan idóneo para cada momento, pero por seguridad actúa transmitiendo un cambio de plan que dirige el controlador hacia la estrategia más parecida a la calculada que ya se encuentre en memoria. Esto obedece a dos propósitos muy importantes:

1. Los planes presentes en el equipo han sido validados por al menos dos personas que poseen la responsabilidad sobre el sistema.
2. Los planes generados no contemplan las condiciones de matriz de conflictos y tiempos intermedios indispensables para garantizar la seguridad de los peatones pero que harían muy costoso desde el punto de vista computacional la ejecución del algoritmo.

Los planes idóneos son almacenados en una lista denominada planes sugeridos, la cual periódicamente debe ser validada, y si se desea programada en el controlador para que esta estrategia de control pueda ser elegida a partir de ese momento. Con el tiempo, se aprovecha la capacidad del controlador de hasta 128 planes de señales y se posee una amplia variedad de estrategias para escoger, preservando la seguridad de los peatones.

El sistema calcula el plan idónea y realiza dos comparaciones: una contra los planes en el equipo (como se explicó previamente) y otra contra los planes sugeridos creados con anterioridad. Si la métrica arroja un valor menor que el umbral para alguno de los planes sugeridos, este se considera ganador y se toma nota del número de veces que ese plan hubiese sido la mejor estrategia en caso de estar programado, dando así una guía a los ingenieros de tráfico sobre su conveniencia de procesarse y cargarse en la memoria del controlador.

MOGUT - Central de Trafico Imatic-RTS

Configuración Ayuda

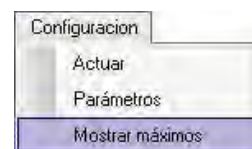
Monitoreo Colas y Planes Actuales Monitoreo Planes Sugeridos

Planes Sugeridos

#	codcruce	vecesganador	dc	datecreated
+	8003	164	24	2013-02-01 10:28:50
+	8003	149	44	2013-02-01 10:28:20
+	3111	126	34	2013-02-12 12:19:47
+	8002	119	26	2013-02-11 17:17:48
+	8003	102	48	2013-02-01 15:18:20
+	8002	101	39	2013-01-31 17:57:12
+	8001	100	33	2013-02-07 13:27:24
+	8001	93	44	2013-02-01 10:28:20
+	8001	92	26	2013-02-01 11:16:22
+	8001	96	78	2013-02-01 15:18:20
+	8004	84	29	2013-02-12 12:17:45
+	8003	81	44	2013-02-04 14:45:22
	numerogrupo	tira	tfv	tfa
	5	0	2	19 22
	6	22	24	41 44
+	8003	71	64	2013-02-04 17:00:01
+	8003	69	44	2013-02-04 14:45:11
+	8001	59	100	2013-01-31 18:01:13
+	8002	52	74	2013-02-01 10:32:51
+	8001	50	57	2013-02-07 13:29:50
+	8003	49	38	2013-02-07 13:27:49
+	8001	45	44	2013-02-04 14:45:11
+	8001	45	122	2013-02-01 16:19:39
+	3111	45	18	2013-02-12 12:21:46
+	8002	43	50	2013-01-31 17:57:42
+	8002	36	41	2013-02-04 15:39:59
+	8001	36	95	2013-01-31 17:59:42
+	8002	35	41	2013-02-04 15:44:31
+	8004	33	47	2013-02-11 17:20:19
+	8004	31	60	2013-02-12 13:33:51
+	8001	27	69	2013-02-01 10:50:21
+	8003	25	41	2013-02-07 13:30:51
+	8003	24	41	2013-02-07 13:28:19
+	8001	24	82	2013-02-01 10:53:22

Actuar

La aplicación permite la visualización de sus procesos de cálculo y selección sin incidir directamente en las intersecciones en campo, cuando la aplicación se inicia la condición por defecto esta deshabilitada y debe activarse desde el menú como se muestra en la imagen.



Una vez ejecutándose, es posible saber si está actuando si cuando la pantalla se refresca (cada 30 segundos) el primer item indica el éxito o fracaso en la transmisión de los planes de señales.

MOGUT - Central de Trafico Imatic-RTS..

Configuracion Ayuda

Monitoreo Colas y Planes Actuales Monitoreo Planes Sugeridos

Registro de eventos

Envío de Planes Exitoso 2013-02-12 12:03:27
El comando de cambio de plan simultaneo ha sido enviado al equipo.

Plan a Enviar 2013-02-12 12:03:27
cruce = 8004 Estructura = 1 Plan = 1 indice = 13

Plan a Enviar 2013-02-12 12:03:27
cruce = 8003 Estructura = 1 Plan = 1 indice = 22

Plan a Enviar 2013-02-12 12:03:27
cruce = 8002 Estructura = 1 Plan = 1 indice = 20

Plan a Enviar 2013-02-12 12:03:27
cruce = 8001 Estructura = 1 Plan = 1 indice = 12

Tiempo de Verde para la Red 2013-02-12 12:03:27
140

Ocupacion Flujo 2013-02-12 12:03:27
cruce = 8004 Flujo = 8 indice Desempeno = 0,26 saturado : NO

Ocupacion Flujo 2013-02-12 12:03:27
cruce = 8004 Flujo = 7 indice Desempeno = 0,18 saturado : NO

Ocupacion Flujo 2013-02-12 12:03:27
cruce = 8003 Flujo = 6 indice Desempeno = 0,56 saturado : NO

Ocupacion Flujo 2013-02-12 12:03:27
cruce = 8003 Flujo = 5 indice Desempeno = 0,06 saturado : NO

Ocupacion Flujo 2013-02-12 12:03:27
cruce = 8002 Flujo = 4 indice Desempeno = 0,11 saturado : NO

Ocupacion Flujo 2013-02-12 12:03:27
cruce = 8002 Flujo = 3 indice Desempeno = 0,06 saturado : NO

Ocupacion Flujo 2013-02-12 12:03:27
cruce = 8001 Flujo = 2 indice Desempeno = 0,06 saturado : NO

Ocupacion Flujo 2013-02-12 12:03:27
cruce = 8001 Flujo = 1 indice Desempeno = 0,39 saturado : NO

Colas

Drag a column header here to group by that column.

id	codcruce	numeroflujo	cola	saturado	datecreated
8001	1	1	0	0	2013-02-12 12:03:14
8001	2	1	0	0	2013-02-12 12:03:14
8002	3	1	0	0	2013-02-12 12:03:14
8002	4	0	0	0	2013-02-12 12:03:14
8003	5	1	0	0	2013-02-12 12:03:14
8003	6	3	0	0	2013-02-12 12:03:14
8004	7	2	0	0	2013-02-12 12:03:14
8004	8	2	0	0	2013-02-12 12:03:14
8001	1	4	0	0	2013-02-12 12:03:04
8001	2	1	0	0	2013-02-12 12:03:04
8002	3	0	0	0	2013-02-12 12:03:04
8002	4	2	0	0	2013-02-12 12:03:04
8003	5	1	0	0	2013-02-12 12:03:04
8003	6	4	0	0	2013-02-12 12:03:04
8004	7	0	0	0	2013-02-12 12:03:04
8004	8	1	0	0	2013-02-12 12:03:04
8001	1	2	0	0	2013-02-12 12:02:54
8001	2	0	0	0	2013-02-12 12:02:54
8002	3	0	0	0	2013-02-12 12:02:54
8002	4	0	0	0	2013-02-12 12:02:54
8003	5	0	0	0	2013-02-12 12:02:54
8003	6	4	0	0	2013-02-12 12:02:54
8004	7	0	0	0	2013-02-12 12:02:54

Planes Actuales

Drag a column header here to group by that column.

id	codigo	numeroestructura	numeroplan	duracionciclo
8001	1	1	6	60
8002	1	1	6	60
8003	1	1	6	60
8004	1	1	6	60



2/F/01/02



NIT. No. 800.195.522-1

GLOSARIO

Malla: Arreglo rectangular de intersecciones viales.

Planes sugeridos: Planes no compatibles con los cargados en el equipo más allá del valor del umbral, quedan latentes en el sistema para ser aprobados por un operador y transmitidos al controlador de tráfico.

Espacio híbrido: Espacio en el que son comparados los planes de señales, generados contra cargados, se denomina híbrido ya que posee elementos tanto de tiempo discreto como de tiempo continuo.